INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

UMI

A Bell & Howell Information Company
300 North Zeeb Road, Ann Arbor MI 48106-1346 USA
313/761-4700 800/521-0600

ERROR ANALYSIS, IMPROVEMENT, AND NEW APPLICATIONS OF NODAL FLUX RECONSTRUCTION METHODS

BY

BRIAN MARK GOLCHERT

B.S., University of Illinois, 1984
M.S., University of Illinois, 1989

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Nuclear Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 1997

Urbana, Illinois

UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

THE GRADUATE COLLEGE

JANUARY 1997

WE HEREBY RECOMMEND THAT THE THESIS BY

BRIAN MARK GOLCHERT

ENTITLED ERROR ANALYSIS, IMPROVEMENT, AND NEW APPLICATIONS OF NODAL FLUX RECONSTRUCTION METHODS

BE ACCEPTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

Abderrafi M. Ougouag
Director of Thesis Research

Barclay G. Jones
Head of Department

Committee on Final Examination†

Abderrafi M. Ougouag
Chairperson

Barclay G. Jones

Kiyoshi Kawahara

James H. Stubbins

† Required for doctor's degree but not for master's.

O-517

ERROR ANALYSIS, IMPROVEMENT, AND NEW APPLICATIONS OF NODAL FLUX RECONSTRUCTION METHODS

Brian Mark Golchert, Ph. D.
Department of Nuclear Engineering
Univeristy of Illinois at Urbana-Champaign, 1997
Dr. Abderrafi M. Ougouag, Advisor

The two types of errors in nodal flux reconstruction are those errors due to the functional interpolation procedure and those errors due to the input data from the nodal code. Bounds for the error due to the interpolation procedure are determined by assuming 'realistic' two dimensional flux shapes and then symbolically propagating these shapes through the mechanisms used to obtain the reconstructed flux. These reconstructed fluxes are then compared to the original, assumed flux to determine an error bound on the interpolation procedure. It has been determined that the functional interpolation procedure will reproduce the assumed flux shape exactly for an assumed flux that is a polynomial of order four or less. The interpolation procedure will exhibit errors if hyperbolic flux shapes are chosen.

The error due to the input data from the nodal code is propagated through the interpolation procedure to determine the effect on the reconstructed flux. This error propagation results in a space-dependent error (i.e. a 2-dimensional error shape) for each assumed error in the nodal input data. The largest errors occur when the input data are currents, particularly the surface average currents. The maximum propagated error is dependent on the material properties of the node and it can be as large as five times the nodal input error.

This methodology for calculating the errors allows an absolute estimate of the 'goodness' of the form of reconstruction. Other methods of flux reconstruction were investigated. The choice of basis functions has relatively little impact on the final form of the reconstructed flux. Any set of polynomial basis functions produces the exact same results while many other choices of basis functions do not admit solutions to the functional interpolation problem. Altering the functional set reveals the major limitation of this analysis: the form of the reconstructed flux has to be mono-energetic. This restricts investigations of potential improvements to a limited set of cases.

Finally, new applications of the reconstructed flux are investigated. One application of the methodology developed in this thesis is the creation of an analytical flux distribution from the region average data from a Monte Carlo code. The specific example of the one-dimensional axial flux in a TRIGA reactor was investigated.

DEDICATION

To my erstwhile mentors:

Polonius, Diogenes, and Don Quixote

ACKNOWLEDGMENTS

I would like to thank Dr. Abderrafi Ougouag for his loyalty, inspiration, consideration and advice during my tenure as a graduate student. Without his considerable talents, this thesis would not have been possible. I would also like to thank my committee members, Dr. M. Ragheb (for my preliminary examination), Dr. J. Stubbins, Dr. B. Jones (for my defense), Dr. R. Uddin (for my defense), and Dr. E. Weinberg for their patience and cooperation.

I truly appreciate the financial support that I have received from the Nuclear Engineering Department, from the Nuclear Reactor Laboratory and from the Department of Energy over my years in graduate school.

I would especially like to thank Dr. Axford and Dr. Jones for their considerable help in expanding my educational frontiers.

Finally, I would like to express my love and gratitude to my wife, Lucia, and to my daughter, Ana, for supporting and caring for me over the last few years.

TABLE OF CONTENTS

1. Introduction....1

2. Background....5
2.a. Neutronics....5
2.b. Functional Interpolation....10

3. Nodal Flux Reconstruction....13
3.a. Koebke and Wagner's Method....13
3.b. Other Methods of Flux Reconstruction....24

4. Methodology to Determine Nodal Flux Reconstruction Errors....26
4.a. Error Due to the Functional Interpolation Method....27
4.a.1. Restricting the Problem....29
4.a.2. Methodology to Determine the Functional Interpolation Error Bound....30
4.a.3. Determination of Errors Associated with 'Realistic' Flux Shapes....31
4.a.3.a. Polynomial Flux Shapes....31
4.a.3.b. Hyperbolic Flux Shapes....37
4.a.3.c. Other Flux Shapes....47
4.a.4. Determination of a Figure of Merit....50
4.b. Error Due to the Nodal Input Data....51
4.b.1. Error Shapes from Nodal Input Data Error....54
4.b.1.a. Error Shape for Node Average Flux Error....55
4.b.1.b. Error Shape for Corner Flux Error....57
4.b.1.c. Error Shape for Corner Current Error....59
4.b.1.d. Error Shape for Surface Averaged Current Error....64
4.b.1.e. Error Shape for Surface Averaged Flux Error....66
4.b.2. Analysis of the Errors Associated with Nodal Input Data Error....67
4.b.3. Determination of the Maximum Errors Due to the Nodal Input Error....68
4.c. Figure of Merit....72

5. Attempts at Reducing the Errors of Nodal Flux Reconstruction....73
5.a. Variation of the Basis Functions....73
5.a.1 Orthogonal Polynomials....74
5.a.2. Hyperbolic Functions....74
5.b. Variation of the Set of Functionals....80
5.c. New Models for Reconstructing the Flux....83
5.d. Conclusions....85

6. New Applications of Nodal Flux Reconstruction....86
6.a. Validation of the Diffusion Approximation....86
6.b. Application of Functional Interpolation to Monte Carlo Results....88
6.b.1. Monte Carlo Methods....88
6.b.2. Location of the Peak Flux....91
6.b.2.a. Location by Linear Interpolation....94
6.b.2.b. Location by Polynomial Interpolation....96
6.b.2.c. Location by Lagrange Interpolation....98
6.b.2.d. Location by Functional Interpolation....101

7. Conclusions and Recommendations....119

Appendix I: Explicit Representations and Graphs of the Errors due to the Nodal Input Data....124
Appendix II: Explicit Representation of the First Derivative of the Reconstructed Flux....146

References....157

Vita....160

LIST OF TABLES

Table 4.1: Deviations for Generic Polynomials of the Type $x^m y^n$ 37

Table 4.2: Deviations for Combinations of Hyperbolic Functions 45

Table 4.3: Total Deviations for Hyperbolic Functions 46

Table 4.4: Deviations from Other Basis Functions 48

Table 4.5: Locations and Maximum Values of the Errors for the Nodal Input Data ... 70-71

Table 5.1: Average Error for Three Values of k 76

Table 5.2: Total Deviations for Three Values of k 77

Table 6.1: Axial Flux Values from the Monte Carlo Code 93

Table 6.2: Locations of Peak Flux from Linear Interpolation 96

Table 6.3: Coefficients for Polynomial Representation of the Fluxes 97

Table 6.4: Calculated Locations in the Core Based on Polynomial Interpolation 98

Table 6.5: Trial Functions for Functional Interpolation 103

Table 6.6: Total Flux Results from Functional Interpolation 105

Table 6.7: Thermal Flux Results from Functional Interpolation 106

LIST OF FIGURES

Figure 3.1: Geometry of a Node....14

Figure 4.1: Error Due to the Node Average Flux....56

Figure 4.2: Error Due to the Corner Flux at (-a,b)....58

Figure 4.3: Error Due to the Corner Flux at (-a,-b)....59

Figure 4.4: Error Due to the x-directed Corner Current at (a,b)....60

Figure 4.5: Error Due to x-directed Corner Current at (-a,b)....61

Figure 4.6: Error Due to x-directed, y-averaged Surface Current at x=a....65

Figure 4.7: Error Due to the x-averaged Surface Flux at y=-b....67

Figure 5.1: Error Shape Due to the y-directed, x-averaged Surface Current at y=b for k=0.1414....78

Figure 5.2: Error Shape Due to the y-directed, x-averaged Surface Current at y=b for k=0.5701....79

Figure 6.1: TRIGA Reactor Core Configuration....90

Figure 6.2: Axial Cross Section of A TRIGA Fuel Element....92

Figure 6.3: Total Flux from Polynomial Interpolation....100

Figure 6.4: Thermal Flux from Polynomial Interpolation....101

Figure 6.5: Total Flux from Functional Interpolation, Trial i....107

Figure 6.6: Total Flux from Functional Interpolation, Trial ii....108

Figure 6.7: Total Flux from Functional Interpolation, Trial iii....109

Figure 6.8: Total Flux from Functional Interpolation, Trial iv....110

Figure 6.9: Total Flux from Functional Interpolation, Trial v....111

Figure 6.10: Total Flux from Functional Interpolation, Trial vi.....112

Figure 6.11: Total Flux from Functional Interpolation, Trial vii.....113

Figure 6.12: Thermal Flux from Functional Interpolation, Trial v.....114

Figure 6.13: Thermal Flux from Functional Interpolation, Trial iv.....115

1. Introduction

When solving an engineering problem, it is often asked: "How good is the computed solution?" To answer this question, the computed solution is compared to an accepted standard or benchmark and then conclusions are drawn depending on how the solution and the standard compare. An ancillary question that is often posed is: "What is the best possible result given this method of solution?" This question is more difficult to answer because the majority of engineering problems are solved using numerical methods. There exist theoretical values for the accuracy of many numerical methods but these theoretical values are not often applicable. For example, the error associated with N point polynomial interpolation is proportional to the Nth derivative of the function being interpolated.[1] Similarly, the error associated with a finite difference approximation to a second order differential equation is on the order of the square of the mesh spacing.[1] These theoretical developments provide an estimate of the error but do not offer any local information concerning the error. What is desired is a methodology that could provide bounds on the error caused by the numerical approximation *and* provide some indication on how the component errors will propagate throughout the region of interest.

Rather than attacking the entire field of numerical analysis, this work will investigate the errors appearing in a particular numerical method (two-dimensional flux reconstruction from the results of a nodal code) in neutronics. Neutronics is the study of the neutron population inside a given volume. In the analysis and design of a nuclear reactor, it is necessary to have a detailed knowledge of the distribution of the neutron

population and of how this neutron density will react to changes in the system. To calculate the neutron density at any point, it is customary to solve the neutron diffusion equation. In one of its most general forms, the neutron diffusion equation is actually a coupled set of N second order differential equations, where N is the number of neutron energy intervals. Given the size (approximately a cube with a side length of fourteen feet) and the varied composition of a nuclear reactor, it is necessary to solve to neutron diffusion equation numerically.

When solving the neutron diffusion equation, two approaches can be taken: solve for spatially detailed neutron flux distributions or solve for region-wide averages of the neutron flux or of the reaction rates. The detailed distribution will indicate how many neutrons are present in a given volume at a given energy while the reaction rates will indicate the rate of production or removal. If the detailed flux has been determined, then the region-wise reaction rates may be obtained by integrating the product of the detailed flux and the material property causing the desired reaction. However, if the reaction rates are initially determined by using nodal methods, then obtaining the detailed neutron flux is a much more complicated matter. The complications arise from the fact that the results from a nodal code can be viewed as one-directional averages of the two-dimensional flux. Thus, the nodal results are functionally dependent on the detailed results not directly dependent on these results. By this statement, it is meant a simple calculation cannot be performed on the nodal results to obtain a detailed distribution. On the other hand, if one had the detailed distribution, then determining the reaction rates is simply an integral of the product of the detailed distribution and the cross section of interest.

Work in this field of reconstructing the detailed flux from nodal results has been on-going for the last fifteen years.[2] Various methods have been investigated for reconstructing the detailed two-dimensional flux from the results of 'one dimensional' nodal methods. These methods utilize a form of functional interpolation of the data provided by the nodal code. The accuracy of these methods is reasonable when compared to established detailed flux distributions from finite difference methods.

A major advantage in determining the reaction rates first and then solving for the detailed flux distribution is speed. Generally, a nodal method will execute much faster than a detailed neutronics method. In the literature, there are estimates that the process of solving for the reaction rates using a nodal code and then reconstructing the detailed flux takes one seventh the time it would take for a finite difference code to determine the detailed flux.[3] However, the main disadvantage of determining the reaction rates and then reconstructing the flux is a loss of accuracy. The proposed work will investigate the limit of the accuracy of published reconstruction methods. Both errors due to the methodology and errors due to the input data from the nodal code will be investigated.

A natural extension of this error analysis is the reduction of these errors. Once the accuracy of the reconstruction schemes has been determined, then methods to improve these schemes will be pursued. There are a number of potential methods to improve the overall accuracy of a reconstructed flux but before these methods can be investigated, a way to compare reconstruction schemes must be formalized. The method of comparison should account for the types of errors that appear in a reconstruction scheme and it should give an indication as to whether a change in a reconstruction

scheme improves the overall results. Once this method to evaluate the 'goodness' of a new reconstruction method has been developed, then the methods for improving the flux reconstruction scheme may be investigated.

Finally, the methodology developed in this work has been applied to other forms of numerical analysis in addition to other topics in the field of nuclear engineering. A few of these applications are presented to demonstrate the versatility of the developed methods, and suggestions are given on how to apply these methods to other problems.

2. Background

The development of successful computational methods requires a thorough understanding of the underlying physical phenomena. In the present work, it is necessary to understand what may happen to a neutron moving within a region of interest. Just as important as the physics is the mathematical formalism required to adequately describe the problem.

2.a. Neutronics

Neutronics is the study of how a neutron, or more accurately how a neutron population, assumed large enough to be treated as a continuum, changes or interacts in a region. The most general description of the neutron population is given by the transport equation.[4]

$$\left[\hat{\Omega}\cdot\vec{\nabla}+\sigma(\vec{r},E)\right]\psi(\vec{r},\hat{\Omega},E)=\int dE'\int d\Omega'\sigma_s(\vec{r},E'\rightarrow E,\hat{\Omega}'\cdot\hat{\Omega})\psi(\vec{r},\hat{\Omega}',E')+\frac{\chi(E)}{k}\int dE'\nu\sigma_f(\vec{r},E')\int d\Omega'\psi(\vec{r},\hat{\Omega}',E') \qquad \text{(Eq. 2.1)}$$

where:

ψ the angular flux which represents the number of neutrons traveling at a specific energy in a specified direction at a particular point;

σ the cross section at a particular point for a given energy;

Ω the direction of travel of the neutron:

E the energy of the neutron:

χ the fraction of neutrons coming out of fission at a given energy;

k the multiplication factor, and

ν the number of neutrons released per fission.

For most core neutronics work, Equation 2.1 is not used, however, because it is very difficult to solve. One source of difficulty is the complex dependence of the material properties (the σ's, $\chi(E)$, and ν) on the space and energy variables (in fact, they are angularly dependent). In addition, the complex geometry of a reactor core often precludes a direct solution to Equation 2.1.

For the analysis of a reactor core, the multigroup diffusion equation is normally used.[5] This equation (actually, a set of coupled equations) can be derived by simplifying the transport equation. Equation 2.2 presents the steady state, multigroup diffusion equation in its most general form:

$$-\nabla \cdot D_g(\bar{r})\nabla\phi_g(\bar{r}) + \Sigma_{Rg}(\bar{r}) = \sum_{g'=1}^{G} \Sigma_{Sg'g}(\bar{r})\phi_{g'}(\bar{r}) + \frac{\chi_g}{k}\sum_{g'=1}^{G}\left\langle \nu\Sigma_f(\bar{r})\right\rangle_{g'}\phi_{g'}(\bar{r})$$

$$g = 1,2....G \qquad \text{(Eq. 2.2)}$$

where:

D is the diffusion coefficient;

ϕ the scalar flux;

Σ is the macroscopic cross section representing the probability per unit path length of an interaction;

Equation 2.2 may be solved directly using finite difference methods.[5] While this method would produce a pointwise representation of the flux ($\phi_g(\bar{r})$), the finite difference method is computationally slow. In addition, to find the reaction rates, the product of the pointwise flux and the cross section of interest needs to be integrated over the domain.

In contrast, nodal methods directly solve for the reaction rates rather than directly solving for the pointwise flux. Although nodal methods are fully three-dimensional, they are illustrated here by considering the two-dimensional form of Equation 2.2, slightly rewritten.[6]

$$-D_g \frac{\partial^2 \phi_g(x,y)}{\partial x^2} - D_g \frac{\partial^2 \phi_g(x,y)}{\partial y^2} + \Sigma_{R,g}\phi_g(x,y) = S_g(x,y) \qquad \text{(Eq. 2.3)}$$
$$g = 1,2....G$$

For Equation 2.3 to be easily solved, it is considered over regions where the material properties are constant. These regions are called nodes. The material in these nodes are either homogeneous initially (that is, only one material) or the cross sections of the various materials are 'homogenized'. The homogenization mathematical procedure produces material properties such that the reaction rates in that node after the homogenization are equivalent to the reaction rates in that node before the homogenization.[7] Additional constraints may be applied during the homogenization procedure (e.g. leakages are conserved). For example, if a typical fuel cell is considered, then the numerical analyst would normally explicitly model the nuclear fuel, the coolant,

the cladding, and any other material present in that cell.[8] This would increase the number of regions to consider and it would further complicate the numerical model. When an assembly, consisting of multiple cells is under consideration, the normal practice is to replace the cells by homogeneous equivalents. In the treatment of a full core, the assemblies are in turn replaced by their homogeneous equivalents. By the use of these homogenized parameters, the material properties are considered to be constant in each assembly, i.e. computational node.

The key idea behind nodal methods is to reduce Equation 2.3 from a two-dimensional equation to two one-dimensional equations. This is accomplished by operating on both sides of Equation 2.3 with:[6]

$$\frac{1}{2a_u}\int_{u=-a_u}^{u=a_u} du \cdot$$

where u may be either x or y and a_u is the half width of the node in the direction u. This operation removes one of the spatial variables from Equation 2.3 and results in the following:

$$-D_g \frac{d^2\bar{\phi}_g(v)}{dv^2} + \Sigma_{R,g}\bar{\phi}_g(v) = \bar{S}_g(v) - L_g(v) \qquad \text{(Eq. 2.4)}$$

where the following definitions have been used:[6]

$$\bar{\phi}_g(v) = \frac{1}{2a_u} \int_{u=-a_u}^{u=a_u} du\, \phi_g(u,v) \ . \quad \text{(Eq. 2.5)}$$

$$\bar{S}_g(v) = \frac{1}{2a_u} \int_{u=-a_u}^{u=a_u} du\, S_g(u,v) \ . \quad \text{(Eq. 2.6)}$$

$$L_g(v) = -\frac{D_g}{2a_u} \frac{\partial \phi_g(u,v)}{\partial u}\bigg|_{u=a_u} + \frac{D_g}{2a_u} \frac{\partial \phi_g(u,v)}{\partial u}\bigg|_{u=-a_u} \ . \quad \text{(Eq. 2.7)}$$

The first definition is referred to as the transverse integrated flux, the second is referred to as the transverse integrated source and the final term is called the transverse leakage. Analogous definitions are applicable in the other direction.

There exist many published methods for solving Equation 2.4.[6,9] The thrust of the present work is not to examine the mechanics of nodal methods but how to manipulate the data available from a nodal code. The information available from a nodal code is related to the transverse integrated properties, not to the two-dimensional flux itself. However, as can be seen in Equations 2.5, 2.6, and 2.7, the nodal information is functionally dependent on the two-dimensional flux. Before discussing how it is possible to take the one-dimensional information from a nodal code and then produce a two-dimensional result, the mathematical formulation of functional interpolation must first be examined.

2.b. Functional Interpolation

When most engineers hear the word 'interpolation', they will immediately think of taking the values of a function at a set of points and constructing an approximate function to fit that set of points. Often this is accomplished with a simple polynomial fit of the data:[10]

$$P(x_i, y_i) = f_i \quad \text{(Eq. 2.8)}$$

Here, f_i is the value of the function at a point (x_i,y_i) and P is a polynomial evaluated at the points in question. As a side note, some of the older numerical analysis literature[11] present polynomial interpolation in one dimension and then state that it is not always possible to extend polynomial interpolation to multiple dimensions. Hammerlin and Hoffmann[1] have shown that multi-dimensional polynomial interpolation may be performed provided that the data points are pairwise distinct.

However, functional interpolation is represented differently:[11]

$$L_i(\phi_g(u, v)) = w_i \quad \text{(Eq. 2.9)}$$

where the L_i are n linear functionals that are independent in the algebraic conjugate space and the w_i are n constants associated with the n functionals.[11] In his treatment of functional interpolation, Davis[11] does not explicitly state that the problem can be

extended to multiple dimensions. Fortunately, Davis does state that the work is being performed in an n-dimensional linear space. This fact, alone, allows one to extend the treatment to an n-dimensional real space.[19]

A quick glance at the two forms for interpolation might tempt a casual observer to conclude that there exists little difference between the two methods. However, this conclusion is erroneous. For 'normal' interpolation, the interpolant is constrained to satisfy only a specific set of points. For functional interpolation, a constant is produced when the functional operates on the entire two-dimensional flux. In this manner, functional interpolation might be considered a global (with regards to the single node) process rather than a pointwise process. The difficulty involved with functional interpolation is the creation of a proper set of functionals.

Functional interpolation is a branch of multivariate approximation theory. A survey paper cites over 1200 papers on the topic of multivariate approximation theory.[12] However, there are very few references that pertain to functional interpolation. Davis, in his book Interpolation and Approximation,[11] provides a very complete mathematical description of functional interpolation. Davis begins with Equation 2.9 and restricts $\phi_g(u,v)$ to be a linear vector space. As was mentioned previously, the set of functionals must be linear and the set must be independent in the algebraic conjugate space. These two terms need to be defined. A linear functional is defined[11] by $L(\alpha x+\beta y)=\alpha L(x)+\beta L(y)$ where α and β are arbitrary constants with x and y assumed to be members of the same linear space (that is, they are linear functions).

A set of functionals that satisfy the following two conditions are considered to be an algebraic conjugate space:[11]

$$\begin{aligned} &(a) \quad (L_1 + L_2)(x) = L_1(x) + L_2(x) \\ &(b) \quad (\alpha L_1)(x) = \alpha L_1(x) \end{aligned} \qquad \text{(Eq. 2.10)}$$

In Equation 2.10, L_1 and L_2 are linear functionals, x is a member of a linear space and α is an arbitrary number.

Equations 2.9 and 2.10 define a functional interpolation problem but these equations do not guarantee that a solution can be found. Davis presents and proves that if the function is in a linear vector space and if the functionals are independent in the algebraic conjugate space, then the functional interpolation problem *may* have a solution if the generalized Gram determinant is not zero.[11] This determinant is represented by:

$$\left|L_i(\phi_j)\right| \neq 0 \qquad \text{(Eq. 2.11)}$$

where the ϕ_j are the basis functions used to approximate the flux. For example, if a polynomial was used to represent the two-dimensional flux, the ϕ_j would be individual basis functions (x, y, xy, x^2y, x^2y^2, etc.). It is this method of functional interpolation that will allow the reconstruction of the two-dimensional flux from the one-dimensional nodal results.

3. Nodal Flux Reconstruction

The preceding chapter presented the necessary framework required to understand nodal flux reconstruction. This chapter will take the one-dimensional results from a nodal code and use functional interpolation to create an approximation to the two-dimensional flux. The historical development of nodal flux reconstruction will be examined to determine what improvements have been introduced, beyond the original method.

3.a. Koebke and Wagner's Method

In a paper published in 1977, Klaus Koebke and Manfred R. Wagner published the first paper to use functional interpolation to reconstruct the two-dimensional flux.[2] The basic ideas presented in their paper have been improved over the years but the core of flux reconstruction is still based on this paper. Koebke and Wagner proposed to represent the two-dimensional flux as a product of two components: the 'smooth' flux and a 'form' factor. The form factor was designed to take into account the variation of material properties in the node. The calculation of these form factors is performed during the material homogenization process[13] and it is assumed that these form factors are known from previous work.

Nodal flux reconstruction essentially entails the creation of the 'smooth' flux. This smooth flux is calculated from the homogenized material parameters. By using

these homogenized parameters, all heterogeneous material variations have been 'smoothed' out. In this manner, the smooth flux can be thought of as a base flux shape. The form factor will represent the effects of heterogeneity so that the product of the smooth flux and the form factor will represent the true flux with the proper magnitude.[14]

Figure 3.1 shows the geometry of a typical node.

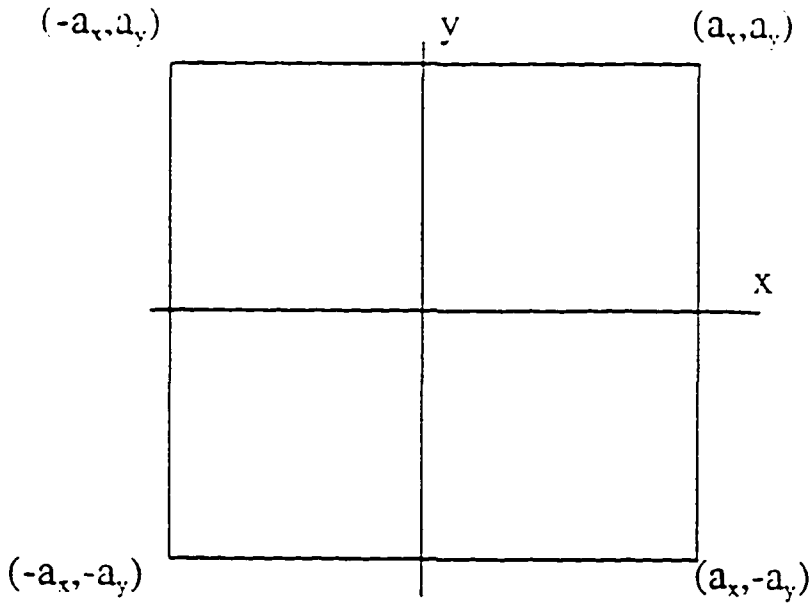

Figure 3.1: Geometry of a Node

Koebke and Wagner created a set of functionals with which to interpolate the nodal data. They chose these twenty-one functionals based on neutronics considerations.[2] The choice of twenty-one functionals mandated a polynomial interpolant that was a fourth order, non-separable polynomial in x and y.

$$\phi_{g,RECON}(x,y)=\sum_{i=0}^{4}\sum_{j=0}^{4}c_{ij}x^{i}y^{j} \qquad \text{(Eq. 3.1)}$$

However, a fourth order, bivariate polynomial has twenty-five terms so Koebke and Wagner set the four highest constants to zero (c_{33} , c_{34} , c_{43} , and c_{44}).

As was stated previously, the physics of the problem dictated the choice of the functionals. The first functional was the node average flux:

$$\left\langle \phi_g(x,y)\right\rangle=\left(\frac{1}{4a_x a_y}\right)\int_{x=-a_x}^{x=a_x}dx\int_{y=-a_y}^{y=a_y}dy\phi_g(x,y) \qquad \text{(Eq. 3.2)}$$

There are also four surface averaged fluxes:

$$\bar{\phi}_g(x=\pm a_x)=\frac{1}{2a_y}\int_{y=-a_y}^{y=a_y}dy\phi_g(x=\pm a_x,y) \qquad \text{(Eq. 3.3)}$$

with two similar equations for the surface averaged flux on the $y=\pm a_y$ faces.

There are four surface averaged currents:

$$\bar{J}_{x,g}(x=\pm a_x)=\frac{-D_g}{2a_y}\int_{y=-a_y}^{y=a_y} dy\left(\frac{\partial \phi_g(x,y)}{\partial x}\right)\Bigg|_{x=\pm a_x} \qquad \text{(Eq. 3.4)}$$

with two similar equations for the surface averaged currents on the $y=\pm a_y$ faces. If the order of the approximation was set at two instead of four, the above set of nine functionals would provide a proper set of functionals for a functional interpolation problem. Unfortunately, this set of nine functionals and a bi-quadratic polynomial basis function does not have a solution.[2] Koebke and Wagner expanded the number of functionals to avoid this situation.

However, by expanding the number of functionals, they 'ran out' of nodal data. The remaining functionals are approximations from the nodal data. Koebke and Wagner devised a method to interpolate the two-dimensional corner point values based on information from the four nodes adjacent to a chosen corner. Thus, the next four functionals describe the two-dimensional flux at each of the four corners:

$$\phi_g(x=\pm a_x, y=\pm a_y)=f(\bar{\phi}_g(x=\pm a_x),\bar{\phi}_g(y=\pm a_y)) \qquad \text{(Eq. 3.5)}$$

The right side of Equation 3.5 represents some method of manipulating the nodal data in order to determine the flux at the corners.[2.15] For this work, it is not necessary to know exactly how these corner point fluxes are determined.

The remaining eight functionals are similar to the corner point fluxes except that four of the functionals relate the x directed current at each of the four corners:

$$-D_g \frac{\partial \phi_g(x,y)}{\partial x}\bigg|_{\substack{x=\pm a_x \\ y=\pm a_y}} = -D_g \frac{\partial}{\partial x} f(\bar{\phi}_g(x), \bar{\phi}_g(y))\bigg|_{\substack{x=\pm a_x \\ y=\pm a_y}} \qquad \text{(Eq. 3.6)}$$

with a similar set of equations for the y directed currents at the corners.

With these definitions, Equations 3.1 through 3.6 constitute a functional interpolation problem. The constants, c_{ij} , in Equation 3.1 are determined by inserting this approximate form of the flux into each of the twenty-one functionals. This will establish a set of twenty-one equations with twenty-one unknowns. Koebke and Wagner solved for these unknowns numerically but it is possible to explicitly solve for these constants by using a symbolic manipulator such as MAPLE.[16]

In order to simplify the equations, the following notation will be used:

$$R1 = \left(\frac{1}{4a_x a_y}\right) \int_{x=-a_x}^{x=a_x} dx \int_{y=-a_y}^{y=a_y} dy \phi_g(x,y)$$

$$R2 = \phi_g(x = a_x, y = a_y)$$

$$R3 = \phi_g(x = -a_x, y = a_y)$$

$$R4 = \phi_g(x = -a_x, y = -a_y)$$

$$R5 = \phi_g(x = a_x, y = -a_y)$$

$$R6 = -D_g \left. \frac{\partial \phi_g(x, y)}{\partial x} \right|_{\substack{x=a_x \\ y=a_y}}$$

$$R7 = -D_g \left. \frac{\partial \phi_g(x, y)}{\partial x} \right|_{\substack{x=-a_x \\ y=a_y}}$$

$$R8 = -D_g \left. \frac{\partial \phi_g(x, y)}{\partial x} \right|_{\substack{x=-a_x \\ y=-a_y}}$$

$$R9 = -D_g \left. \frac{\partial \phi_g(x, y)}{\partial x} \right|_{\substack{x=a_x \\ y=-a_y}}$$

$$R10 = -D_g \left. \frac{\partial \phi_g(x, y)}{\partial y} \right|_{\substack{x=a_x \\ y=a_y}}$$

$$R11 = -D_g \left. \frac{\partial \phi_g(x, y)}{\partial y} \right|_{\substack{x=-a_x \\ y=a_y}}$$

$$R12 = -D_g \left. \frac{\partial \phi_g(x, y)}{\partial y} \right|_{\substack{x=-a_x \\ x=-a_x}}$$

$$R13 = -D_g \left. \frac{\partial \phi_g(x, y)}{\partial y} \right|_{\substack{x=a_x \\ x=-a_x}}$$

$$R14 = \frac{-D_g}{2a_y} \int_{y=-a_y}^{y=a_y} dy \left. \frac{\partial \phi_g(x, y)}{\partial x} \right|_{x=a_x}$$

$$R15 = \frac{-D_g}{2a_y} \int_{y=-a_y}^{y=a_y} dy \left. \frac{\partial \phi_g(x, y)}{\partial x} \right|_{x=-a_x}$$

$$R16 = \frac{-D_g}{2a_x} \int_{x=-a_x}^{x=a_x} dx \left. \frac{\partial \phi_g(x, y)}{\partial y} \right|_{y=a_y}$$

$$R17 = \frac{-D_g}{2a_x} \int_{x=-a_x}^{x=a_x} dx \left. \frac{\partial \phi_g(x, y)}{\partial y} \right|_{y=-a_y}$$

$$R18 = \frac{1}{2a_y} \int_{y=-a_y}^{y=a_y} dy \phi_g(x = a_x, y)$$

$$R19 = \frac{1}{2a_x} \int_{y=-a_y}^{y=a_y} dy \phi_g(x = -a_x, y)$$

$$R20 = \frac{1}{2a_y} \int_{x=-a_x}^{x=a_x} dx \phi_g(x, y = a_y)$$

$$R21 = \frac{1}{2a_y} \int_{x=-a_x}^{x=a_x} dx \phi_g(x, y = -a_y)$$

These twenty-one pieces of nodal data will be stored in the array $\bar{R}$. The polynomial basis functions will be stored in an array:

$$\bar{F} = \left(1, y, y^2, y^3, y^4, x, xy, xy^2, xy^3, xy^4, x^2, x^2y, x^2y^2, x^2y^3, x^2y^4, x^3, x^3y, x^3y^2, x^4, x^4y, x^4y^2\right)$$

while the unknown coefficients, c_{ij}, will be stored in an array, $\bar{C}$, in an order corresponding to that of the basis functions. In this manner, the flux may be written as:

$$\phi_g(x, y) = \bar{C} \cdot \bar{F}^T \qquad \text{(Eq. 3.7)}$$

Applying the twenty-one constraints to the chosen form of the flux results in the following matrix equation:

$$A\bar{C} = \bar{R} \quad \text{(Eq. 3.8)}$$

Here, the matrix A results from having the twenty-one functionals operate on the chosen form of the flux. The product of the inverse of A and the vector R will result in symbolic forms for the unknown coefficients, using the assumed form of the reconstructed flux as was given in Equation 3.1:

$$c_{00} = \frac{1}{64D_g}\begin{pmatrix} 216D_gR1 + 10D_gR2 + 10D_gR3 + 10D_gR4 + 10D_gR5 + a_xR6 - a_yR7 \\ -a_xR8 + a_xR9 + a_yR10 + a_yR11 - a_yR12 - a_yR13 - 6a_yR14 + 6a_xR15 \\ -6a_yR16 + 6a_yR17 - 48D_gR18 - 48D_gR19 - 48D_gR20 - 48D_gR21 \end{pmatrix}$$

$$c_{01} = \frac{-1}{32a_yD_g}\begin{pmatrix} 9D_gR2 + 9D_gR3 - 9D_gR4 - 9D_gR5 + a_xR6 - a_xR7 + a_xR8 \\ -a_xR9 + 2a_yR10 + 2a_yR11 + 2a_yR12 + 2a_yR13 - 12a_yR16 \\ -12a_yR17 - 42D_gR20 + 42D_gR21 \end{pmatrix}$$

$$c_{02} = \frac{-3}{64a_y^2D_g}\begin{pmatrix} 132D_gR1 + 13D_gR2 + 13D_gR3 + 13D_gR4 + 13D_gR5 + a_xR6 \\ -a_xR7 - a_xR8 + a_xR9 + 2a_yR10 + 2a_yR11 - 2a_yR12 - 2a_yR13 \\ -2a_xR14 + 2a_xR15 - 12a_yR16 + 12a_yR17 - 26D_gR18 \\ -26D_gR19 - 66D_gR20 - 66D_gR21 \end{pmatrix}$$

$$c_{03} = \frac{1}{16a_y^3D_g}\begin{pmatrix} D_gR2 + D_gR3 - D_gR4 - D_gR5 + a_yR10 + a_yR11 + \\ a_yR12 + a_yR13 - 6a_yR16 - 6a_yR17 - 6D_gR20 + 6D_gR21 \end{pmatrix}$$

$$c_{04} = \frac{5}{64a_x^4 D_g}\begin{pmatrix} 36D_g R1 + 3D_g R2 + 3D_g R3 + 3D_g R4 + 3D_g R5 + a_x R10 \\ +a_x R11 - a_x R12 - a_x R13 - 6a_x R16 + 6a_x R17 - 6D_g R18 \\ -6D_g R19 - 18D_g R20 - 18D_g R21 \end{pmatrix}$$

$$c_{10} = \frac{-1}{32a_x D_g}\begin{pmatrix} 9D_g R2 - 9D_g R3 - 9D_g R4 + 9D_g R5 + 2a_x R6 + 2a_x R7 \\ +2a_x R8 + 2a_x R9 + a_x R10 - a_x R11 + a_x R12 - a_x R13 \\ -12a_x R14 - 12a_x R15 - 42D_g R18 + 42D_g R19 \end{pmatrix}$$

$$c_{11} = \frac{1}{8a_x a_x D_g}\begin{pmatrix} 4D_g R2 - 4D_g R3 + 4D_g R4 - 4D_g R5 + a_x R6 + a_x R7 \\ -a_x R8 - a_x R9 + a_x R10 - a_x R11 - a_x R12 + a_x R13 \end{pmatrix}$$

$$c_{12} = \frac{3}{16a_x a_y^2 D_g}\begin{pmatrix} 6D_g R2 - 6D_g R3 - 6D_g R4 + 6D_g R5 + a_x R6 + a_x R7 \\ +a_x R8 + a_x R9 + a_y R10 - a_y R11 + a_y R12 - a_y R13 \\ -2a_x R14 - 2a_x R15 - 12D_g R18 + 12D_g R19 \end{pmatrix}$$

$$c_{13} = \frac{-1}{8a_x a_y^3 D_g}\left(D_g R2 - D_g R3 + D_g R4 - D_g R5 + a_y R10 - a_y R11 - a_y R12 + a_y R13\right)$$

$$c_{14} = \frac{-5}{32a_x a_y^4 D_g}\begin{pmatrix} 3D_g R2 - 3D_g R3 - 3D_g R4 + 3D_g R5 + a_y R10 \\ -a_y R11 + a_y R12 - a_y R13 - 6D_g R18 + 6D_g R19 \end{pmatrix}$$

$$c_{20} = \frac{-3}{64a_x^2 D_g}\begin{pmatrix} 132D_g R1 + 13D_g R2 + 13D_g R3 + 13D_g R4 + 13D_g R5 + 2a_x R6 \\ -2a_x R7 - 2a_x R8 + 2a_x R9 + 2a_y R10 + 2a_y R11 - 2a_y R12 \\ -2a_y R13 - 12a_x R14 + 12a_x R15 - 2a_y R16 + 2a_y R17 \\ -66D_g R18 - 66D_g R19 - 26D_g R20 - 26D_g R21 \end{pmatrix}$$

$$c_{21} = \frac{3}{16a_x^2 a_y}\begin{pmatrix} 6D_g R2 + 6D_g R3 - 6D_g R4 - 6D_g R5 + a_y R6 - a_y R7 + a_y R8 - a_y R9 \\ +a_y R10 + a_y R11 + a_y R12 + a_y R13 - 2a_y R16 - 2a_y R17 - 12D_g R20 + 12D_g R21 \end{pmatrix}$$

$$c_{22} = \frac{9}{32a_x^2 a_y^2 D_g}\begin{pmatrix} 32D_g R1 + 8D_g R2 + 8D_g R3 + 8D_g R4 + 8D_g R5 + a_y R6 \\ -a_y R7 - a_y R8 + a_y R9 + a_y R10 + a_y R11 - a_y R12 - a_y R13 \\ -2a_y R14 + 2a_y R15 - 2a_y R16 + 2a_y R17 - 16D_g R18 \\ -16D_g R19 - 16D_g R20 - 16D_g R21 \end{pmatrix}$$

$$c_{23} = \frac{-3}{16a_x^2 a_y^3 D_g}\begin{pmatrix} D_g R2 + D_g R3 - D_g R4 - D_g R5 + a_y R10 + a_y R11 + a_y R12 \\ +a_y R13 - 2a_y R16 - 2a_y R17 - 2D_g R20 + 2D_g R21 \end{pmatrix}$$

$$c_{24} = \frac{-15}{64a_x^2 a_y^4 D_g}\begin{pmatrix} 12D_g R1 + 3D_g R2 + 3D_g R3 + 3D_g R4 + 3D_g R5 + a_y R10 + a_y R11 - a_y R12 \\ -a_y R13 - 2a_y R16 + 2a_y R17 - 6D_g R18 - 6D_g R19 - 6D_g R20 - 6D_g R21 \end{pmatrix}$$

$$c_{30} = \frac{1}{16a_x^3 D_g}\begin{pmatrix} D_g R2 - D_g R3 - D_g R4 + D_g R5 + a_x R6 + a_x R7 + a_x R8 \\ +a_x R9 - 6a_x R14 - 6a_x R15 - 6D_g R18 + 6D_g R19 \end{pmatrix}$$

$$c_{31} = \frac{-1}{8a_x^3 a_y D_g}\left(D_g R2 - D_g R3 + D_g R4 - D_g R5 + a_x R6 + a_x R7 - a_x R8 - a_x R9\right)$$

$$c_{32} = \frac{-3}{16a_x^3 a_y^2 D_g}\begin{pmatrix} D_g R2 - D_g R3 - D_g R4 + D_g R5 + a_x R6 + a_x R7 + a_x R8 \\ +a_x R9 - 2a_x R14 - 2a_x R15 - 2D_g R18 + 2D_g R19 \end{pmatrix}$$

$$c_{40} = \frac{5}{64 a_1^4 D_g} \begin{pmatrix} 36 D_g R1 + 3 D_g R2 + 3 D_g R3 + 3 D_g R4 + 3 D_g R5 \\ + a_1 R6 - a_1 R7 - a_1 R8 + a_1 R9 - 6 a_1 R14 + 6 a_1 R15 \\ - 18 D_g R18 - 18 D_g R19 - 6 D_g R20 - 6 D_g R21 \end{pmatrix}$$

$$c_{41} = \frac{-5}{32 a_1^4 a_1 D_g} \begin{pmatrix} 3 D_g R2 + 3 D_g R3 - 3 D_g R4 - 3 D_g R5 + a_1 R6 \\ - a_1 R7 + a_1 R8 - a_1 R9 - 6 D_g R20 + 6 D_g R21 \end{pmatrix}$$

$$c_{42} = \frac{-15}{64 a_1^4 a_1^2 D_g} \begin{pmatrix} 12 D_g R1 + 3 D_g R2 + 3 D_g R3 + 3 D_g R4 + 3 D_g R5 \\ + a_1 R6 - a_1 R7 - a_1 R8 + a_1 R9 - 2 a_1 R14 + 2 a_1 R15 \\ - 6 D_g R18 - 6 D_g R19 - 6 D_g R20 - 6 D_g R21 \end{pmatrix}$$

Equation 3.1 with these coefficients constitutes the functional form of the reconstructed flux. The results from the nodal code along with the material and geometric properties of the node now completely describe the unknown coefficients.

3.b. Other Methods of Flux Reconstruction

The majority of the published methods have used the reconstruction formulation of Koebke and Wagner. Over the years, a few improvements have been developed. In 1984, Koebke and Hetzelt improved the reconstruction of the thermal flux by replacing the polynomial basis functions with hyperbolic basis functions.[17] They also presented a component of the thermal flux that uses the reconstructed fast flux in addition to the hyperbolic basis functions.

Besides the change in the basis functions, other works have only made small deviations from Koebke and Wagner's work. These changes pertain to how to approximate the corner values[18] or which basis functions to use for the thermal flux.[17,15] However, the basic model is still the one developed in 1977. It is this model that will be initially analyzed for the error contributions to the final result.

4. Methodology to Determine Nodal Flux Reconstruction Errors

This section of this thesis will determine a methodology for analytically calculating the errors associated with two-dimensional nodal flux reconstruction. As was mentioned previously, the errors associated with nodal flux reconstruction were historically determined by a comparison with more detailed two-dimensional calculations. One aim of this work is to establish a procedure for determining the magnitude and the causes of the errors in nodal flux reconstruction without explicitly doing a comparison with the detailed calculation. If such a methodology could be determined, then it could be analytically stated which reconstruction method is superior.

Before proceeding, it is prudent to consider the types of errors that can be expected from a functional interpolation problem. Hildebrand states that there are always three types of errors associated with any numerical problem:[10] errors associated with the method, input errors, and errors due to roundoff. Given the machine precision available today, it is assumed that roundoff error is insignificant when compared to the contribution of the two other error components. By analytically (symbolically) formulating and solving the problems of errors, it is not necessary to be concerned with roundoff affecting the results of this work. Each of the two remaining errors will be investigated and quantified.

4.a. Error Due to the Functional Interpolation Method

Two-dimensional nodal flux reconstruction is an approximation of the exact two-dimensional solution. Since this is an approximation, it is possible to use the established tools for error analysis from approximation theory. Very little work has been done in the field of error determination for functional interpolation problems. One reference presents the following remainder formula for a functional interpolation problem:[11]

$$x_R = \frac{\begin{vmatrix} x & x_1 & \dots & x_n \\ L_1(x) & L_1(x_1) & \dots & L_1(x_n) \\ \dots & \dots & \dots & \dots \\ L_n(x) & L_1(x_1) & \dots & L_n(x_n) \end{vmatrix}}{\begin{vmatrix} L_1(x_1) & L_1(x_2) & \dots & L_1(x_n) \\ \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ L_n(x_1) & L_n(x_2) & \dots & L_n(x_n) \end{vmatrix}} \qquad \text{(Eq. 4.1)}$$

A remainder is simply the original function minus the approximating function. A remainder of zero indicates that the approximation perfectly reproduces the original function.

In this remainder formula, x is the function being approximated and the x_i's are the basis functions used to approximate the original function. As an example in one dimension, the neutron flux may be written as an expansion in Legendre polynomials:

$$\Phi(z) = \sum_{i=0}^{N} c_i P_i(z) \qquad \text{(Eq. 4.1)}$$

For this example, $\Phi(z)$ would be the function being approximated (corresponding to x in the general formulation) and the Legendre polynomials would be the basis functions used in the approximation (the counterparts of the x_i).

The $L_i(\ )$ terms in Equation 4.1 are the functionals involved in the interpolation problem. Explicit representations of these functionals were presented in the previous chapter. It can be readily seen that the determination of the remainder requires knowledge of the original function. In fact, an investigation of the field of errors (or remainders) associated with approximation theory reveals that all of the remainder formulae will require knowledge of the form of the function being approximated.[19] This is a major handicap since the two-dimensional flux is not generally known.

It should be noted that the remainder formula presented above assumes that there is no error associated with the functional data. It may be rewritten to include these errors (x_i replace by $x_i+\delta x_i$) but, as written, this remainder may be considered to be the remainder associated with the functional interpolation methodology. The determinant in the numerator of this formula provides a function (in this case, a two-dimensional function) since the function being approximated and the basis functions used in the chosen form of the approximation appear in this determinant. Unfortunately, this remainder formula does not provide a number. It produces a function that describes the remainder at each point. It is desired to have a number with which to compare the 'goodness' of a nodal flux reconstruction.

4.a.1 Restricting the Problem

The remainder formula presents a concise equation for determining the error. It is, however, not practical. If a reactor physicist knew the form of the two-dimensional flux distribution before performing the nodal calculation, then there would not be a need for these calculations as the sought for answer would already be available. Thus, the first step in determining the errors must be to find a method to quantify those errors due to the methodology. In order to find these errors, it is necessary to use "engineering knowledge" to restrict the problem.

Nodal flux reconstruction tries to model the smooth (homogeneous) flux. Since the flux is relatively smooth, it is not expected to have many undulations, rapid variations, nor many inflection points. The non-homogeneous flux details would have been absorbed in the shape, or form, factor as was explained in Chapter 3. It seems reasonable to assume that the smooth flux can be represented by a low order bivariate polynomial or perhaps a set of hyperbolic/trigonometric basis functions. This assumption can be further justified by examining the homogenized two-dimensional diffusion equation for a node:

$$-D_g \nabla^2 \Phi_g(x,y) + \Sigma_{R,g} \Phi_g(x,y) = S_g(x,y) \quad \text{. (Eq. 4.3)}$$

If strictly polynomial solutions to the above equation are examined (that is, the two-dimensional flux is considered to be a finite ordered polynomial expansion in both x and

in y), then it is readily seen that the order of the polynomial will be restricted by the order of the approximation of the source term. The maximum order of a power series solution will not exceed the highest order present in the differential equation. It is common to approximate the transverse integrated one-dimensional nodal source term by a quadratic polynomial, so it should be possible to approximate the two-dimensional source by a bi-quadratic polynomial.[6]

The above discussion simply restricts the possible forms for realistic two-dimensional fluxes. Still, a workable method to determine the magnitude of the error due to the reconstruction methodology is not at hand. What is proposed is not to look for an exact form for the error but rather to determine an upper error bound associated with the nodal flux reconstruction procedure. This approach is reasonable since a reconstruction procedure may encounter a wide variety of fluxes to be reconstructed. Thus, trying to determine an "exact" error is a unrealistic quest. Instead, it should be attempted to determine the limits of the method; hence, the search for error bounds.

4.a.2 Methodology to Determine the Functional Interpolation Error Bound

The method proposed here is very simple: Consider a series of analytic forms for "reasonable" two-dimensional flux shapes, and from these flux shapes create the nodal input data for the functional interpolation procedure. As was stated previously, the nodal input data are functionals related to the two-dimensional flux. The nodal data of the reconstruction process consist of information extracted from the two-dimensional flux,

such as the flux evaluated at the corner points and flux derivatives or integrals. Since an analytic form for the two-dimensional flux is chosen, all of the nodal input data may be calculated symbolically, resulting in explicit forms for each input item.

With this symbolic input data, a reconstructed flux is calculated and then compared with the original assumed flux. The difference between the assumed flux and the reconstructed flux will be the absolute error associated with the functional interpolation procedure. An illustrative way of looking at this analysis is to assume that the reconstruction procedure is a black box. A known function is put into the black box resulting in a reconstructed flux. This known function is then compared with what comes out of the black box. Analyzing the results from this procedure should allow the quantification of the error bounds associated with the functional interpolation procedure.

4.a.3. Determination of Errors Associated with Sample Realistic Flux Shapes

4.a.3.a. Polynomial Flux Shapes

Realistic flux shapes, representative of shapes expected in actual applications, need to be chosen for testing the reconstruction method. It was originally proposed to investigate a series of simple flux shapes. An example of one of these bivariate shapes is a linear function in x and a quadratic function in y,

$$\Phi(x,y) = (c_1 + c_2 x)(c_3 + c_4 y + c_5 y^2) = a_1 + a_2 x + a_3 y + a_4 xy + a_5 y^2 + a_6 xy^2 \quad . \quad \text{(Eq. 4.4)}$$

After performing a variety of these analyzes, it became apparent that there exists a simpler method of addressing this part of the problem. By definition, the functionals involved in the interpolation procedure are linear. Thus, the result of a functional operating on a function could be added to the result of the same functional operating on a different function. Therefore, instead of inputting specific forms for a flux, the individual contributions of the products of the basis functions will be investigated. For the case presented in Equation 4.4, the functional interpolation could be performed on the entire function or it could be performed individually on the six basis functions (for this example, the basis functions are: 1, x, y, xy, xy^2, and y^2) and then the results added together. By analyzing the functionals operating on the basis functions, the problem is simplified.

This analysis could be further simplified if this procedure would be performed on "generic" polynomials; polynomials with arbitrary powers in x and in y. By using these simplifications, only four analyzes need to be performed in determining the error due to the functional interpolation procedure. The reason four analyzes need to be performed instead of one is that the functionals involve integrals and derivatives of functions. Thus, the results will differ if the exponents of x or of y will be odd or even. With these four results with arbitrary exponents, the functional interpolation error associated with any polynomial function can be determined by summing the components of each of the basis functions.

To perform this analysis, the form of the assumed flux is taken to be:

$$\Phi_{TRUE}(x,y) = cx^m y^n \quad . \text{(Eq. 4.5)}$$

Where $-a \le x \le +a$ and $-b \le y \le +b$. The dimension 'a' is the half width of the node in the x direction and the dimension 'b' is the half width of the node in the y direction. To simplify the calculations later, the form of the assumed flux is rewritten as:

$$\Phi_{TRUE}(\eta,\xi) = ca^m b^n \eta^m \xi^n \quad \text{(Eq. 4.6)}$$

Where $\eta = x/a$ and $\xi = y/b$. After choosing the parity of the exponents m and n, the form of the assumed flux is inserted into the definitions of the nodal input data to get symbolic forms for the input data. These symbolic data are then inserted into the interpolation procedure to obtain a reconstructed flux. The four reconstructed fluxes obtained following the procedures shown in Chapter 3 (Section a) are, for m and n odd,

$$\Phi_{RECON}(\eta,\xi) = \frac{-1}{2} ca^m b^n \left[\eta^3\xi(1-m) + \eta\xi^3(1-n) - \eta\xi(4-m-n)\right] , \quad \text{(Eq. 4.7)}$$

for m odd and n even,

$$\Phi_{RECON}(\eta,\xi) = \frac{ca^m b^n}{8(n+1)} \left[\begin{array}{l} \eta(-8n+12+n^2-4m+2mn) + \eta\xi^4(5n^2-10n) + \\ \eta\xi^2(-6n^2+30n-6mn) + \eta^3(-2mn+4m+2n-4) + \eta^3\xi^2(6mn-6n) \end{array} \right]$$

(Eq. 4.8)

for m even and n odd,

$$\Phi_{RECON}(\eta,\xi)=\frac{ca^m b^n}{8(m+1)}\left[\begin{array}{l}\xi(m^2+2mn-8m-4n+12)+\xi\eta^2(30m-6m^2-6mn)+\\ \xi\eta^4(5m^2-10m)+\xi^3(-4+2m+4n-2mn)+\xi^3\eta^2(6mn-6m)\end{array}\right].$$

(Eq. 4.9)

and for m even and n even.

$$\Phi_{RECON}(\eta,\xi)=\frac{-ca^m b^n}{16(m+1)(n+1)}\left[\begin{array}{l}(-8mn+12m+12n-2n^2-2m^2+m^2n+n^2m-16)+\\ \eta^2(-48m+30mn-3n^2m-6m^2n+12m^2)+\\ \eta^2\xi^2(-108mn+18n^2m+18m^2n)+\eta^2\xi^4(30mn-15n^2m)+\\ \eta^4(20m-10mn-10m^2+5m^2n)+\eta^4\xi^2(30mn-15m^2n)+\\ \xi^2(30mn-48n+12n^2-6n^2m-3m^2n)+\\ \xi^4(20n-10mn-10n^2+5n^2m)\end{array}\right]$$

(Eq. 4.10)

The above formulae allow comparisons between any chosen bivariate polynomial and the reconstructed flux resulting from that choice. An absolute error can be established by the following simple definition:

$$\varepsilon_{abs}=\Phi_{TRUE}(\eta,\xi)-\Phi_{RECON}(\eta,\xi) \quad . \quad \text{(Eq. 4.11)}$$

By systematically checking all reasonable combinations of m and of n, it is quickly discovered that the low order bivariate polynomials are exactly reproduced by the functional interpolation procedure. By low order, it is meant that the sum of the exponents, m and n, is less than or equal to five except when m or n is equal to zero. In this case, the results are exact only up to a fourth order polynomial.

Of interest is the average absolute error associated with each of these reconstructed fluxes. For this work, the average absolute error will be defined as:

$$\langle \varepsilon_{ABS} \rangle = \frac{\int_{-1}^{1} d\eta \int_{-1}^{1} d\xi (\varepsilon_{ABS})}{\int_{-1}^{1} d\eta \int_{-1}^{1} d\xi} \quad \text{. (Eq. 4.12)}$$

Using this definition and inserting each of the four forms for the reconstructed flux, the average error for each of four reconstructed flux shapes is equal to zero regardless of the order or the parity of either m or n.

This is not to imply that the functional interpolation procedure will produce a perfectly reconstructed flux every time. In order to better visualize the deviation from a perfect reconstruction, it is necessary to use a norm, such as the integral average of the square of the absolute error. This figure of merit will give an indication of the total deviation of the reconstructed flux from the assumed shape. This is found by simply integrating the square of the absolute error over the entire region of the node. This value is then divided by the integral of the square of the true flux. This is done to remove any dependencies on the node size. (The dimensions of the node appear in the representations for the absolute error.) Finally, the square root of the result is taken to find the total deviation from the true flux.

$$\sigma = \sqrt{\langle \varepsilon_{ABS}^2 \rangle} = \sqrt{\frac{\dfrac{\int_{-1}^{1} d\eta \int_{-1}^{1} d\xi \left(\varepsilon_{ABS}\right)^2}{\int_{-1}^{1} d\eta \int_{-1}^{1} d\xi}}{\left(\dfrac{\int_{-1}^{1} d\eta \int_{-1}^{1} d\xi \left(\Phi_{TRUE}(\eta,\xi)\right)^2}{\int_{-1}^{1} d\eta \int_{-1}^{1} d\xi} \right)}} \quad \text{(Eq. 4.13)}$$

Table 4.1 presents the value of the total deviation (or "relative area") associated with each combination of exponents. By "relative area", it is meant that if both the true and the reconstructed fluxes were plotted on the same graph, these numbers would represent the space between the two curves normalized to the integral square of the original function. If there is no space between the curves, then these curves lie one on top of the other: they are the same.

One of the goals of this work is to determine a method to appraise the "goodness" of a flux reconstruction method. The values in Table 4.1 should give a good indication of how well the functional interpolation procedure performs for polynomial functions. This topic of "goodness" will be discussed later in this chapter.

	m=0	m=1	m=2	m=3	m=4	m=5	m=6	m=7	m=8	m=9
n=0	0.000	0.000	0.000	0.000	0.000	0.637	0.285	1.665	0.791	2.983
n=1	0.000	0.000	0.000	0.000	0.000	0.637	0.285	1.665	0.791	2.983
n=2	0.000	0.000	0.000	0.000	0.000	0.637	0.285	1.665	0.791	2.983
n=3	0.000	0.000	0.000	0.533	0.279	1.757	0.900	3.523	1.824	5.703
n=4	0.000	0.000	0.000	0.279	0.146	1.143	0.566	2.485	1.250	4.176
n=5	0.637	0.637	0.637	1.757	1.143	3.524	2.088	5.969	3.390	8.871
n=6	0.285	0.285	0.285	0.900	0.566	2.088	1.169	3.802	2.068	5.924
n=7	1.665	1.665	1.665	3.523	2.485	5.969	3.802	9.020	5.494	12.60
n=8	0.791	0.791	0.791	1.824	1.250	3.390	2.068	5.494	3.199	8.045
n=9	2.983	2.983	2.983	5.703	4.176	8.871	5.924	12.60	8.045	16.87

Table 4.1: Deviations for Generic Polynomials of the Type $x^m y^n$

4.a.3.b. Hyperbolic Flux Shapes

The previous section investigated polynomial shapes. There are, however, other flux shapes that should be investigated. The most logical choice of basis functions for the solution of the diffusion equation would be hyperbolic functions. If it is assumed that the x and y solutions can be represented by a series of hyperbolic sines and cosines, then the total flux representation would be a non-separable product solution with various combinations of hyperbolic functions:

$$\Phi_g(x,y) = \sum_{i=0}^{M}\sum_{j=0}^{N} c_{ij} f_i(k_i x) g_j(k_j y) \qquad \text{(Eq. 4.14)}$$

where the c's are constants to be determined and the f's and g's are either hyperbolic sines or cosines. For example, Koebke and Hetzelt[17] have written their thermal flux as a non-separable product of hyperbolic functions times the fast flux:

$$\Phi_2(x,y) = \Phi_1(x,y) \sum_{i,j=0}^{4} c_{ij} F_i(x) F_j(y)$$

where:

$$F_0(x) = 1 \qquad F_1(x) = \sinh(kx) \qquad F_2(x) = \cosh(kx)$$

$$F_3(x) = \sinh(2kx) \qquad F_4(x) = \cosh(2kx)$$

with a similar set of basis functions for the y direction.

The summation limits in Equation 4.14 may appear to be arbitrary since it is possible to have differing arguments inside the hyperbolic functions. However, the summation is limited by the number of chosen constraints for the given method of reconstruction. For this part of the thesis, all of the combinations of hyperbolic sines and cosines that satisfied the number of original constraints were considered. For brevity,

three simplified combinations were investigated: $\cosh(k_1x)\cosh(k_2y)$, $\sinh(k_1x)\sinh(k_2y)$, and $\cosh(k_1x)\sinh(k_2y)$. It is assumed that the x's and the y's are interchangeable. These three combinations of hyperbolic functions are versatile in that the values of k_1 and k_2 are undetermined. Thus, with a choice of the two k's, one could have any combination of products of hyperbolic functions (through the utilization of various hyperbolic identities).

With this representation of the flux, the method that was used in determining the total deviation from the generic polynomials is repeated. By using the same form of the absolute error as was defined in Equation 4.11, the integral averaged absolute error is found to be zero for all combinations of hyperbolic functions. Note that it was not possible to present reasonable representations similar to Equations 4.7 through 4.10 for the reconstructed fluxes. Even with simple assumed fluxes such as a bivariate polynomial, the symbolic form for the flux is quite extensive.

Proceeding directly with the calculation of the total deviation presents a problem. The definition of the total deviation took the integral of the square of the absolute error and divided it by the integral of the square of the true flux. This definition was used to remove any dependence on either material properties or on geometry for the particular case of reconstruction using bi-variate polynomial basis functions. Removing these dependencies proved impossible with the forms for the deviation from hyperbolic basis functions due to the presence of the node size and the material properties in the arguments of the hyperbolic functions. For example, using Equation 4.13, the total deviation from a true flux representation of:

$$\Phi_{TRUE}(x,y) = f\cosh(k_1 x)\sinh(k_2 y) \quad \text{(Eq. 4.15)}$$

is:

$$\Xi^2 = \frac{-1}{4725}\begin{pmatrix}
1360800k_2^2b^2 + 6804000a^2k_1^2 + 160a^4b^8k_2^8k_1^4 + 288a^6b^6k_1^6k_2^6 \\
+1814400k_2^4b^4a^2k_1^2 - 3175200a^2b^2k_1^2k_2^2 + 42525k_2^5A_4a^3b^5k_1^3 \\
-151200k_2^5A_4a^2b^5k_1^2 + 340200A_2b^4a^3k_1^3k_2^4 + 151200k_2^5A_2a^2b^5k_1^2 \\
+680400A_4b^4a^2k_1^2k_2^4 + 3402000A_2ba^2k_1^2k_2 + 226800e^{(2k_1a)}b^2a^2k_1^2k_2^2 \\
-1360800e^{(-2k_2b)}b^3a^2k_1^2k_2^3 - 6804000e^{(-2k_2b)}ba^2k_1^2k_2 \\
-1360800e^{(-2k_1a)}b^2ak_1k_2^2 - 18900a^3b^6k_2^6k_1^3e^{(-2k_1a)} \\
+6804000e^{(2k_2b)}ba^2k_1^2k_2 - 5216400e^{(2k_2b)}b^2a^2k_1^2k_2^2 \\
+1360800e^{(2k_2b)}b^3a^2k_1^2k_2^3 + 3288600A_3a^2b^2k_1^2k_2^2 \\
+1360800A_2ab^3k_1k_2^3 - 42525A_3k_2^5a^3b^5k_1^3 + 151200k_2^5e^{(-2k_2b)}a^2b^5k_1^2 \\
-151200k_2^5e^{(2k_2b)}a^2b^5k_1^2 + 680400A_2b^2ak_1k_2^2 \\
-1360800A_3b^3a^3k_1^3k_2^3 - 2948400A_3b^2k_1^3k_2^2 + 680400A_2ab^4k_1k_2^4 \\
+18900a^3b^6k_2^6k_1^3e^{(2k_1a)} + 2041200A_2b^3a^2k_1^2k_2^3 + 226800k_2^5A_2ab^5k_1 \\
+3402000A_3ba^2k_1^2k_2 - 5216400e^{(-2k_2b)}b^2a^2k_1^2k_2^3 \\
-226800k_2^5A_3ab^5k_1 + 680400A_2b^4a^2k_1^2k_2^4 - 226800k_2^5A_1ab^5k_1 \\
-151200k_2^5A_1a^2b^5k_1^2 - 42525k_2^5A_1a^3b^5k_1^3 - 1360800e^{(2k_1a)}ab^4k_1k_2^4 \\
-1360800A_1b^3ak_1k_2^3 - 1360800A_3b^3ak_1k_2^3 - 680400A_3b^2ak_1k_2^2 \\
+2041200A_3b^3a^2k_1^2k_2^3 - 907200e^{(2k_1a)}b^2a^3k_1^3k_2^2 \\
+1360800e^{(2k_1a)}b^2ak_1k_2^2 + 907200e^{(-2k_1a)}b^2a^3k_1^3k_2^2 \\
+3402000A_4ba^3k_1^3k_2 + 680400A_1b^2ak_1k_2^2 - 2041200A_1b^3a^2k_1^2k_2^3 \\
-340200A_3a^3b^4k_1^3k_2^4 + 1360800A_2b^3a^3k_1^3k_2^3 \\
+3402000A_2ba^3k_1^3k_2 + 3288600A_2b^2a^2k_1^2k_2^2 \\
+226800e^{(-2k_1a)}b^2a^2k_1^2k_2^2 - 3402000A_1ba^3k_1^3k_2
\end{pmatrix}$$

$$\frac{-1}{4725}\left(\begin{array}{l}
+2948400A_1b^2a^3k_1^3k_2^2-1360800A_1b^3a^3k_1^3k_2^2+2948400A_2b^2a^3k_1^3k_2^2\\
+226800k_2^5A_4ab^5k_1-144b^5a^6k_1^6k_2^5A_4-80b^8a^4k_1^4k_2^8e^{(2l\cdots}\\
+72a^6b^6k_1^6k_2^6A_3+288b^5a^6k_1^6k_2^5e^{(-2k_2b)}+144b^5a^6k_1^6k_2^5A_3\\
+1728b^5a^5k_1^5k_2^5A_2+72a^6b^6k_1^6k_2^6A_4+144a^6b^6k_1^6k_2^6e^{(2k_2b)}\\
+144a^6b^6k_1^6k_2^6e^{(-2k_2b)}+72b^7a^5k_2^7k_1^5A_4+936a^5k_1^5b^6k_2^6A_2\\
-8328b^6k_2^6a^4k_1^4e^{(-2k_2b)}+4164b^6k_2^6a^4k_2^4A_4+72a^6k_1^6b^6k_2^6A_2\\
+864a^6k_1^6b^4k_2^4e^{(-2k_2b)}-936a^5k_1^5b^6k_2^6A_4+4104b^4k_2^4a^5k_1^5A_1\\
+40b^8k_2^8a^4k_1^4A_1+144a^6k_1^6b^6k_2^6e^{(-2k_1a)}-1728b^5k_2^5a^5k_1^5A_3\\
-6984a^4k_1^4b^5k_2^5A_1+40b^8k_2^8a^4k_1^4A_4+1584a^5k_1^5b^6k_2^6e^{(-2k_1a)}\\
-25344a^4k_1^4b^4k_2^4e^{(2k_1a)}-1344a^4k_1^4b^7k_2^7e^{(-2k_2b)}+72a^6b^6k_1^6k_2^6A_1\\
+432a^6k_1^6b^4k_2^4A_1-680400A_4b^2k_2^2ak_1+1360800A_4b^3k_2^3ak_1\\
-3402000A_3bk_2a^3k_1^3-3402000A_4bk_2a^2k_1^2+3288600A_4b^2k_2^2a^2k_2^2\\
-2041200A_4b^3k_2^3a^2k_1^2+453600e^{(2k_1a)}b^4k_2^4a^2k_1^2+680400A_3b^4k_2^4a^2k_1^2\\
-680400A_3b^4a^2k_2^4k_1^2+1360800e^{(-2k_1a)}b^4k_2^4ak_1+151200A_3k_2^5b^5a^2k_1^2\\
+340200A_1a^3k_1^3b^4k_2^4+453600e^{(-2k_1a)}b^4k_2^4a^2k_1^2+680400A_1b^4k_2^4a^2k_1^2\\
-680400A_4b^4k_2^4ak_1+42525A_2b^5k_2^5a^3k_1^3-226800e^{(-2k_1a)}b^4k_2^4a^3k_1^3\\
+226800e^{(2k_1a)}a^3k_1^3b^4k_2^4-340200A_4a^3k_1^3b^4k_2^4\\
-2948400A_4b^2k_2^2a^3k_1^3+1360800A_4b^3k_2^3a^3k_1^3-3402000A_1ba^2k_1^2k_2\\
+4164A_3b^6k_2^6a^4k_1^4+4104A_2b^4k_2^4a^5k_1^5+936A_1a^5k_1^5b^6k_2^6\\
+4164A_1b^6k_2^6a^4k_1^4-72A_3b^7k_2^7a^5k_1^5-4104A_3b^4k_2^4a^5k_1^5\\
-936A_3a^5k_1^5b^6k_2^6+40A_3b^8k_2^8a^4k_1^4+672A_3a^4k_1^4b_7k_2^7+4164A_2b^6k_2^6a^4k_1^4\\
+432A_4a^6k_1^6b^4k_2^4-672A_1a^4k_1^4b^7k_2^7-1728A_1b^5k_2^5a^5k_1^5+432A_3a^6k_1^6b^4k_2^4\\
+8208e^{(2k_1a)}b^4k_2^4a^5k_1^5+72A_2b^7k_2^7a^5k_1^5+12672A_3a^4k_1^4b^4k_2^4\\
+5640e^{(-2k_1a)}b^6k_2^6a^4k_1^4-1584e^{(2k_1a)}a^5k_1^5b^6k_2^6-288e^{(2k_2b)}b^5k_2^5a^6k_1^6\\
-72A_1b^7k_2^7a^5k_1^5-80e^{(2k_2b)}b^8k_2^8a^4k_1^4+12672A_2a^4k_1^4b^4k_2^4+144A_2b^5k_2^5a^6k_1^6\\
-25344e^{(-2k_2b)}a^4k_1^4b^4k_2^4-864e^{(2l_1a)}a^6k_1^6b^4k_2^4-8328e^{(2k_2b)}b^6k_2^6a^4k_1^4\\
+144e^{(2k_1a)}a^6k_1^6b^6k_2^6+4932e^{(-2k_2b)}a^4k_1^4b^5k_2^5-864e^{(-2k_1a)}a^6k_1^6b^4k_1^4\\
+6984A_3a^4k_1^4b^5k_2^5+12672A_1a^4k_1^4b^4k_2^4-25344e^{(2k_2b)}a^4k_1^4b^4k_2^4\\
+12672A_4a^4k_1^4b^4k_2^4-1360800b^4k_2^4-6984A_4a^4k_1^4b^5k_2^5\\
-4932e^{(2k_2b)}a^4k_1^4b^5k_2^5+5640e^{(2k_1a)}b^6k_2^6a^4k_1^4+864e^{(2k_2b)}a^6k_1^6b^4k_2^4\\
-4104A_4b^4k_2^4a^5k_1^5+1728A_4a^5k_1^5b^5k_2^5-25344e^{(-2k_1a)}a^4k_1^4b^4k_2^4
\end{array}\right)$$

$$
\frac{-1}{4725}\Big(
+1344a^4b^7k_2^7k_1^4e^{(2k_2b)}+672a^4k_1^4b^7k_2^7A_2+40b^8k_2^8a^4k_1^4A_2-144b^5k_2^5a^7k_1^7A_1
-672a^4k_1^4b^7k_2^7A_4-80b^8k_2^8a^4k_1^4e^{(-2k_2b)}-80b^8k_2^8a^4k_1^4e^{(-2k_1a)}
-3402000a^2k_1^2e^{(2k_1a)}+1701000a^2k_1^2A_4+1701000a^2k_1^2A_3
-3402000a^3k_1^3e^{(-2k_1a)}-3402000a^2k_1^2e^{(-2k_2b)}+1701000a^2k_1^2A_1
-1701000a^3k_1^3A_3-1701000a^3k_1^3A_4+1701000a^2k_1^2A_2+3402000a^3k_1^3e^{(2k_1a)}
-3402000a^2k_1^2e^{(-2k_1a)}-3402000a^2k_1^2e^{(2k_2b)}+1701000a^3k_1^3A_1
+680400b^3k_2^3A_2-680400b^2k_2^2e^{(2k_2b)}+1360800b^3k_2^3e^{(2k_2b)}
-680400b^2k_2^2e^{(2k_1a)}+340200b^2k_2^2A_4+226800b^5k_2^5e^{(2k_2b)}
-226800b^5k_2^5e^{(-2k_2b)}-113400b^5k_2^5A_1-680400b^4k_2^4e^{(-2k_2b)}
-680400b^4k_2^4e^{(2k_2b)}+113400b^5k_2^5A_3+113400b^5k_2^5A_2-113400b^5k_2^5A_4
+680400b^4k_2^4e^{(2k_1a)}+340200b^4k_2^4A_4+680400b^4k_2^4e^{(-2k_1a)}+340200b^4k_2^4A_3
+340200b^4k_2^4A_1+340200b^4k_2^4A_2-680400b^3k_2^3A_4-1360800b^3k_2^3e^{(-2k_2b)}
+680400b^3k_2^3A_3-680400b^3k_2^3A_1+340200b^2k_2^2A_1+340200b^2k_2^2A_3
-680400b^2k_2^2e^{(-2k_1a)}-680400b^2k_2^2e^{(-2k_2b)}+340200b^2k_2^2A_2
+6984a^4k_1^4b^5k_2^5A_2+432a^6k_1^6b^4k_2^4A_2+3288600a^2k_1^2b^2k_2^2A_1
+680400ak_1b^4k_2^4A_1-1728a^6k_1^6b^4k_2^4-86880a^4k_1^4b^6k_2^6+50688a^4k_1^4b^4k_2^4
\Big)
$$

$$
\sigma^2=\Xi^2\Big/a^3k_1^3b^5k_2^5\left(4ak_1-e^{(2k_1a)}+e^{(2k_1a)}\right)\left(4bk_2+e^{(2k_2b)}-e^{(-2k_2b)}\right)
$$

where:

$$
A_1=e^{(2k_2b-2k_1a)},
$$

$$
A_2=e^{(-2k_1a-2k_2b)},
$$

$$
A_3=e^{(2k_1a)-2k_2b)},
$$

$$A_4 = e^{(2k_1a-2k_2b)}.$$

(Eq. 4.16)

Equation 4.16 gives some indication of the complexity of the terms. Each term in Equation 4.16 is dependent on the material (k_1 and k_2) and on the geometry (a and b).

As a side note, some of the terms in Equation 4.16 may be combined to form hyperbolic functions. The total deviation is represented in the above manner due to the limitations of MAPLE. By converting all hyperbolic functions representations into exponential representations, a lot of computation time and a significant amount of storage space may be saved. It appears that it is easier for the program to store and to manipulate exponential functions rather than hyperbolic functions. As an example, the file that stored the expression for the deviation of:

$$\Phi_g(x,y) = f\cosh^2(k_1x)\cosh^2(k_2y) \quad \text{(Eq. 4.17)}$$

in hyperbolic form contained approximately 150 kilobytes. This is to be compared with the file that stored the same expression for the deviation of Equation 4.17 converted to exponential form. The file that stored the deviation in exponential form contained only 85 kilobytes.

Finding a numerical expression for the total deviations will require some estimate of the constant k. A symbolic value of k may be found by examining the form of the diffusion equation (Eq. 4.3) and then defining k as:

$$k = \sqrt{\frac{\Sigma_{r,g}}{D_g}} \quad \text{(Eq. 4.18)}$$

Thus, for a given material composition, the value of the constant may be determined. Several values of k were investigated by using the materials from a common numerical benchmark problem (IAEA2D). The values for the total deviation associated with various combinations of hyperbolic basis functions (using the constants from IAEA2D) are presented in Table 4.2. A first estimate of how 'good' the functional interpolation procedure approximates the hyperbolic basis functions would be the summation of the deviations for all of the chosen values of the constant. This option has a very significant drawback. This estimate of 'goodness' would only be applicable to that particular set of materials investigated (that particular numerical problem).

Trial function	k=0.1414	k=0.1826	k=0.4472	k=0.4610	k=0.5701
cosh(kx)	1.4207	1.8619	36.0830	42.4048	149.1537
sinh(kx)	1.7457	2.1496	36.1673	42.4823	149.1917
$\cosh(k_1x)\cosh(k_2y)$	0.2147	0.3794	2.1374	2.2445	3.1171
$\cosh(k_1x)\sinh(k_2y)$	0.2147	0.3794	2.1374	2.2445	3.1171
$\sinh(k_1x)\sinh(k_2y)$	1.7517	1.6414	2.6679	2.7569	3.5135

Table 4.2: Deviations for Combinations of Hyperbolic Functions

Ideally, this estimate of "goodness" should be independent of the material composition. If the "goodness" estimate were independent of the material properties then this estimate could be used to compare various reconstruction methods to determine the best overall method. However, using Equation 4.13 results in a total deviation that is dependent on the constant k. This deviation that is dependent on k is then integrated over a range of the constants

$$\sigma^2 = \frac{\int_{k=0.1}^{k=0.7} dk\sigma^2(k)}{\int_{k=0.1}^{k=0.7} dk} \quad \text{(Eq. 4.19)}$$

The choice of the integral limits will be guided by 'expected' or 'reasonable' values for this constant. The lower limit was chosen to be 0.1 and the upper limit, 0.7, based on the five values of the constant from the IAEA numerical problem. Any values

of the constant smaller than the lower limit imply materials with very large diffusion lengths or very small removal cross sections. For these regions, the diffusion approximation would probably break down. Likewise, for values of the constant greater than the upper limit, it is assumed that there is a very strong absorber present in the region (high removal cross section and a small diffusion length). This is another region where the diffusion approximation is expected to fail.

Equation 4.19 will give a representation of the 'goodness' for any form of the functional interpolation problem. Again, this integral is normalized over the range of the constants. This is done to ensure consistency when comparisons are to be made with the bi-variate polynomial reconstruction method.

Table 4.3 presents the total deviation as calculated from Equation 4.18 for the various hyperbolic basis functions examined.

Basis function	Total deviation
cosh(kx)	432.8481
sinh(kx)	432.9596
$\cosh(k_1x)\cosh(k_2y)$	3.9870
$\cosh(k_1x)\sinh(k_2y)$	3.9870
$\sinh(k_1x)\sinh(k_2y)$	4.3042

Table 4.3: Total Deviations for Hyperbolic Functions

The deviations for the one-dimensional basis functions (cosh(kx) and sinh(kx)) were two orders of magnitude greater than the bi-variate basis functions. After confirming these results by repeating the calculations, it was postulated that these large errors were due to the hyperbolic basis function having only one dimension. However, a single-variable basis function ($\cosh^2(kx)$) was found to have a total deviation on the same order of magnitude (1.8985) as the two-dimensional basis functions. The reason why the single-variable hyperbolic basis functions have such markedly different total deviations is not understood.

4.a.3.c. Other Flux Shapes

Of interest is how the functional interpolation procedure will approximate various functions that are not necessarily 'solutions' to the neutron diffusion equation. This analysis will give some indication on the flexibility of the methodology. If the total deviations are relatively small for a variety of basis functions, then the method is considered flexible. However, if the method adequately approximates some basis functions but poorly approximates other basis functions, then the method is clearly not flexible. Table 4.4 shows the deviations for a number of different basis functions. The deviations presented in this table represent the total deviation (as defined by Equation 4.19) and the deviations associated with a representative number of the constants from the IAEA numerical problem.

Those basis functions that contain two constants were evaluated by assuming that both constants were the same. The symbolic analysis was performed using two unknown constants. By using two constants instead of one, greater flexibility was achieved. The resulting forms for the deviations are the most general possible.

Basis function	k=0.1414	k=0.4472	k=0.5701	Total deviation
cos(kx)	2.1574	728.7515	9746.0	27,030.6
sin(kx)	2.3557	767.4598	8990.1	26,920.0
$\cos(k_1x)\cos(k_2y)$	2.7550	1004.9556	14,374.0	38,004.0
$\cos(k_1x)\sin(k_2y)$	2.2763	1058.4738	13,259.1	38,454.0
$\sin(k_1x)\sin(k_2y)$	0.0128	1.2887	1.3613	1.0915
$\cos^2(kx)$	0.1258	0.6100	0.7519	0.6933
$\sin^2(kx)$	0.1058	0.6543	0.6743	0.6895
e^{kx}	1.2962	25.5541	105.4834	306.1101
xcos(kx)	1.1084	124.2900	1832.40	4952.6
ycos(kx)	0.7965	126.9610	1688.32	4681.8
xsin(kx)	1.0406	135.309	1465.6	4434.6
ysin(kx)	1.0661	132.932	1557.12	4662.7
$\exp(k_1x)\exp(k_2y)$	0.8879	1.3755	1.7830	2.1735
xcos(ky)	1.0534	126.227	1688.05	4681.8
xsin(ky)	1.0661	132.932	1557.12	4662.7

Table 4.4: Deviations from Other Basis Functions

As can be seen by the results presented in Table 4.4, there is no consistency in the total deviation from the functional interpolation procedure. For a few basis functions, the methodology gives very good results (for example, the square of trigonometric terms) but other functions give extremely poor results (cos(kx) and sin(kx)). Coupling the results from Tables 4.3 and 4.4, it can be concluded that this form of the flux reconstruction is not very flexible. Of course, the basis functions used in Table 4.4 are not 'expected' in neutronics (with the caveat that hyperbolic functions can be rewritten as exponentials). However, even the results for hyperbolic basis functions indicate that the methodology is not very flexible. The difference in total deviations of two orders of magnitude give ample evidence for this conclusion.

With all of these calculations concerning the error due to the functional interpolation method, it should be possible to arrive at an upper bound. The term upper bound is used in a loose sense in that it represents the worst functional interpolation error associated with a realistic flux shape. Clearly, by examining Tables 4.1 and 4.3, the single, most dominant functional interpolation error is due to the simplest hyperbolic basis functions: sinh(kx) and cosh(kx). The errors associated with these two basis functions are two order of magnitude greater than any other case. Thus, the upper bound will be set at 432.9596 as this is the worst case for the realistic basis functions.

4.a.4. Determination of a Figure of Merit

This leads directly to the question previously asked: 'How good is this method at reconstructing (approximating) the flux?' To answer this question, it is necessary to recall that "...the problem of best approximation amounts to the minimization of a distance..."[2]. Even this quotation requires some clarification. Does best imply the smallest distance for ANY approximation to the function or does best imply the smallest maximum absolute error for a given set of functions (that is, the smallest worst case)? Since the range of expected functions is unknown, the worst case is also unknown. To make the problem tractable, the set of basis functions will be limited to those that are most probably expected in reactor physics applications: polynomials and hyperbolics. Again, since the constant for the hyperbolics is also not known ahead of time, the total deviation over a range of the constants will be used. Thus, the results presented in Table 4.1 and 4.3 can be manipulated to give a representation of how 'good' this form of flux reconstruction is.

Summing all of the deviations in Tables 4.1 and 4.3 results in a number that may be used as a gauge when comparing other functional interpolation methods. This total represents the total deviation for all expected basis functions. If this total could be lowered by a different functional interpolation method, then this new method could be considered as an improvement over the current method. The sum of the entries in Tables 4.1 and 4.3 is 1,114.4 and it is dominated by the contributions from cosh(kx) and sinh(kx).

4.b. Error Due to the Nodal Input Data

Now that the error bound for the functional interpolation procedure has been determined, those errors associated with the nodal input data will be examined. This is accomplished by assuming that there is no error associated with the functional interpolation procedure. Then, the nodal input data is perturbed and symbolically manipulated through the functional interpolation procedure. The resulting expression will be regrouped into a term involving the unperturbed nodal input data and a term that is associated with the perturbation. This term associated with the perturbation will be the absolute error associated with that piece of nodal input data.

Before beginning with a mathematical formulation for the absolute error due to the nodal input data, a proper representation for the reconstructed flux needs to be formulated. If the form of the reconstructed flux is defined as:

$$\Phi_g(x,y)=\sum_{i=0}^{m}\sum_{j=0}^{n}c_{ij}f_m(x)g_n(y) \quad \text{(Eq. 4.20)}$$

then the set of functionals operate on this function to create the constraints necessary to solve for the c_{ij}. This set of constraints may be expressed as:

$$\mathbf{A}\bar{\mathbf{c}}=\bar{\mathbf{r}} \quad \text{(Eq. 4.21)}$$

Here, the column vector **c** stores the c_{ij}, in a pre-defined order while the column vector $\bar{\mathbf{r}}$ stores the nodal input data in the same order as the column vector $\bar{\mathbf{c}}$. For this particular case, **A** is the square constraint matrix that is obtained when the form of the reconstructed flux is inserted into the functionals. Thus, the sixth row of the matrix **A** will contain the coefficients for the unknown constants, c_{ij}, that result when the sixth functional is applied to the assumed form of the reconstructed flux. The sixth entry in the column vector $\bar{\mathbf{r}}$ will be the sixth piece of nodal data used to constrain the flux. It should be noted that it is possible to reconstruct the two-dimensional flux with a constraint matrix that is not square by using a generalized inverse of the constraint matrix.[20] The constraint matrix would not be square if the number of functionals was not the same as the number of unknown coefficients. The situation where there exist more coefficients than functionals is referred to as an under-determined system. The system is considered to be over-determined if there are more functionals than coefficients. Finally, just selecting enough functionals to equal the number of unknown coefficients does not guarantee a solution. The set of functionals must be linearly independent in the algebraic conjugate space as was described in Chapter 2 (Section b).

To find the complete expression for the unknown coefficients, it is necessary to multiply both sides of the matrix equation by the inverse of the constraint matrix:

$$\bar{\mathbf{c}} = \mathbf{A}^{-1}\bar{\mathbf{r}} \qquad \text{(Eq. 4.22)}$$

The validity of Equation 4.22 depends on the existence of the inverse of the constraint matrix. This inverse will not exist if the constraint matrix is singular. This may occur if the set of functionals are poorly chosen. For example, flux reconstruction using a bi-variate second order polynomial flux representation will have a singular constraint matrix if the set of functionals consists of the node average flux, the four surface average fluxes, and the four surface averaged currents.[13]

Finally, if $\bar{f}^T$ is defined as the row vector which stores the bivariate polynomial basis functions in the same order as the unknown coefficient vector, then the reconstructed flux will be represented as:

$$\Phi_g(x,y) = \bar{f}^T \cdot \bar{c} = \bar{f}^T \cdot \mathbf{A}^{-1}\bar{r} \quad \text{(Eq. 4.23)}$$

This formulation allows the errors due to the nodal input data to be mathematically modeled. Assume that the i^{th} piece of nodal data has an error and the other nodal input data are perfect. This perturbation will appear in the $\bar{r}$ vector:

$$\Phi_{g,RECON}(x,y) = \bar{f} \cdot \bar{c}_{RECON} = \bar{f} \cdot \mathbf{A}^{-1}\bar{r}_{RECON} = \bar{f} \cdot \mathbf{A}^{-1}(\bar{r}_{TRUE} - \delta r_i) \quad \text{(Eq. 4.24)}$$

The subscripts RECON and TRUE correspond to those values from the reconstructed flux and from the true (assumed) flux, respectively. The term, δr_i, represents the error in the i^{th} piece of nodal data. Without this error, the reconstruction procedure is assumed to

perfectly model the assumed flux. By referring to the definition of absolute error, it is seen that:

$$\varepsilon_{ABS} = \Phi_{g,TRUE}(x, y) - \Phi_{g,RECON}(x, y) = \bar{\mathbf{f}} \cdot \bar{\mathbf{c}}_{TRUE} - \bar{\mathbf{f}} \cdot \bar{\mathbf{c}}_{RECON} \quad \text{(Eq. 4.25)}$$

$$\varepsilon_{ABS} = \bar{\mathbf{f}} \cdot \bar{\mathbf{c}}_{TRUE} - \bar{\mathbf{f}} \cdot \left(\bar{\mathbf{c}}_{TRUE} - \mathbf{A}^{-1}\delta\mathbf{r}_i\right) = \bar{\mathbf{f}} \cdot \mathbf{A}^{-1}\delta\mathbf{r}_i \quad \text{(Eq. 4.26)}$$

The total error due to the entire set of the nodal input data would be the sum of each of the components due to the individual pieces of input data. Upon examination of the form for the absolute error, it is evident that there exists a two-dimensional shape associated with each piece of nodal input data since the vector, $\bar{\mathbf{f}}$, contains the polynomial basis functions. Analysis of these shapes will reveal the maximum, average and spatial variation of these errors. Appendix I contains the explicit, analytical expressions for the shapes associated with each of the twenty-one pieces of nodal input data.

4.b.1. Error Shapes from Nodal Input Data Error

Even though there are twenty-one pieces of input data, there are only five different shapes for all the errors. Any of the twenty-one error shapes may be found by rotating one of these five basic shapes. These shapes are derived from the inputs for the

node average flux, the corner flux, the x-directed corner current, the y-directed corner current, the surface average currents, and the surface average flux. Figure 4.1 depicts the error shape for the node average input. This is the most 'regular' shape of all the errors associated with the input data. Presenting these shapes properly does pose a few problems. While it is important to visualize the general shape of the error propagation, a representation of the error should also indicate location of the maximum, the minimum, and the inflection points. The location of the maximum in Figure 4.1 is obvious but it is difficult to perceive that there exist four minima (local minima) just inside each of the four corners. The magnitude of the function at these inflection points is very small when compared to the scale of the plot, and thus these point are obscured.

4.b.1.a. Error Shape for Node Average Flux Error

The analytic form for this error is:

$$\frac{1}{4ab}\int_{-a}^{a} dx \int_{-b}^{b} dy \phi(x,y) \Rightarrow r_1 = \frac{1}{64}\begin{pmatrix} 216 - 396\xi^2 - 396\eta^2 + 180\xi^4 + 180\eta^4 \\ -180\xi^4\eta^2 - 180\xi^2\eta^4 + 576\xi^2\eta^2 \end{pmatrix} \quad \text{(Eq. 4.27)}$$

where the functional definition of the node average flux is on the left of the arrow and the shape is to the right of the arrow. The shape has been converted from (x,y) coordinates

to ($\eta = x/a, \xi = y/b$) coordinates to simplify the writing of the equation. Also, it more clearly shows a point raised below.

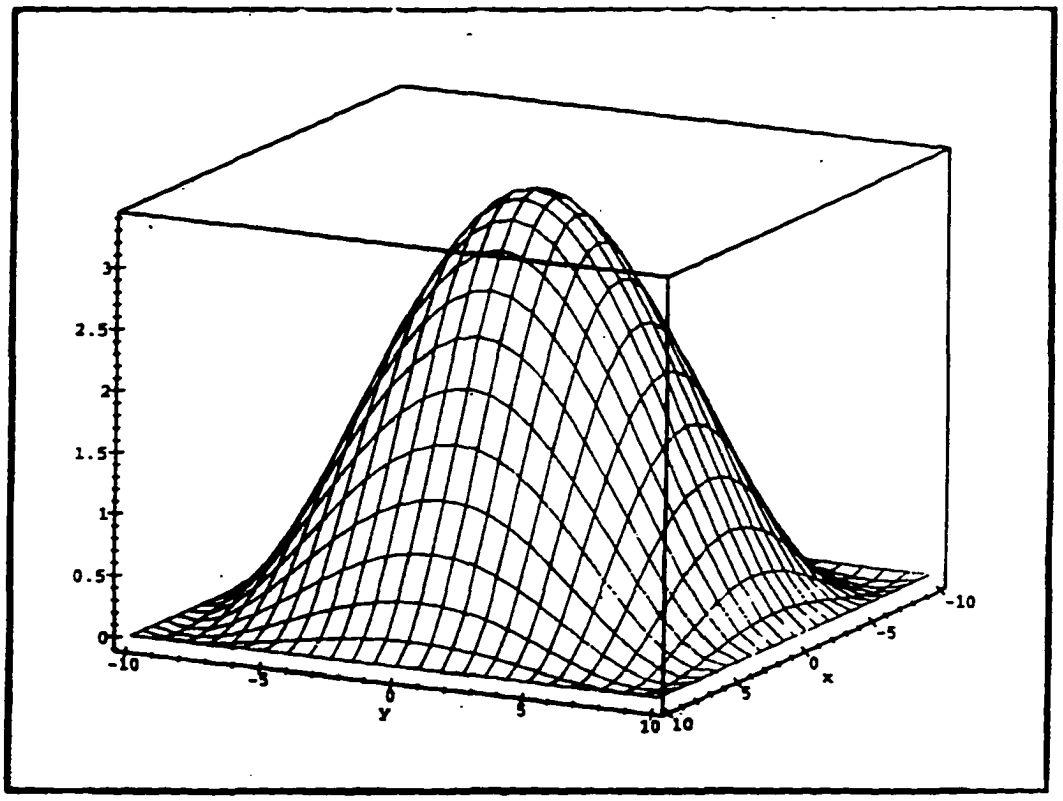

Figure 4.1: Error Due to the Node Average Flux

Before proceeding, the physical significance of what this plot represents will be discussed. Suppose that the node average flux had an error and that this error could be represented by δr_i . This error could be a number or a percent error but it is probably easier to visualize if the error is considered in percent. A δr_i percent error will propagate throughout the node with zero percent error on the edges and 3.375 δr_i percent error at

the center of the node. This shape implies that the effect of this error on the reconstructed flux is not constant but depends on the location in the node. It is also dependent on the method of reconstruction but for this method of reconstruction. it is independent of the physical size of the node. It is incorrect to assume that it is also independent of the material properties of the node. This assumption of no dependency on the material properties is only true for those nodal input which are not currents.

4.b.1.b. Error Shape for Corner Flux Error

To illustrate the rotation of the flux shape to obtain all twenty-one input shapes, consider the shape (Figure 4.2) associated with the corner flux at (-a,b). The equation for this error shape is given by:

$$\phi(-a,b) \Rightarrow r_3 = \frac{1}{64}\begin{pmatrix} 30\eta\xi^4 - 4\eta^3 - 39\xi^2 - 32\eta\xi + 4\xi^3 + 8\eta^3\xi - 30\eta^4\xi - 72\eta\xi^2 \\ +15\eta^4 + 72\eta^2\xi + 15\xi^4 - 39\eta^2 - 18\xi + 12\eta^3\xi^2 + 18\eta - 45\eta^4\xi^2 \\ +144\eta^2\xi^2 + 8\eta\xi^3 + 10 - 45\eta^2\xi^4 - 12\eta^2\xi^3 \end{pmatrix}$$

(Eq. 4.28)

To determine the shape associated with the corner flux at (-a,-b), Figure 4.2 could be rotated ninety degrees counterclockwise about the point 33(x=0,y=0) to obtain Figure 4.3. Alternatively, the value of $\xi = -\xi$ could be inserted into Equation 4.28 to obtain

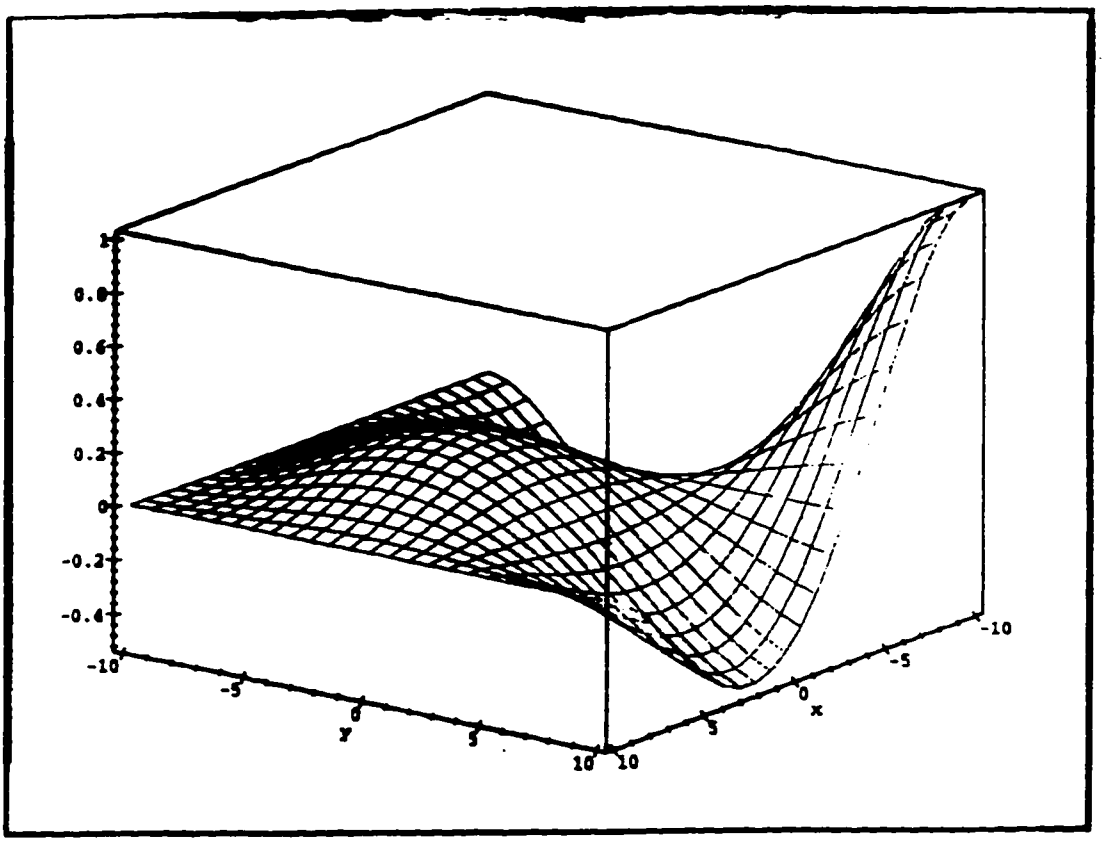

Figure 4.2: Error Due to the Corner Flux at (-a,b)

the resulting expression (see Appendix 1) for the error due to the corner flux at (-a,-b).

An examination of these error plots for the corner fluxes reveals that the propagated error never exceeds the initial value of the error. This fact is not true for the node average flux. Also, the maximum error for the corner fluxes will occur at the corner in question and will quickly dissipate further away from the corner.

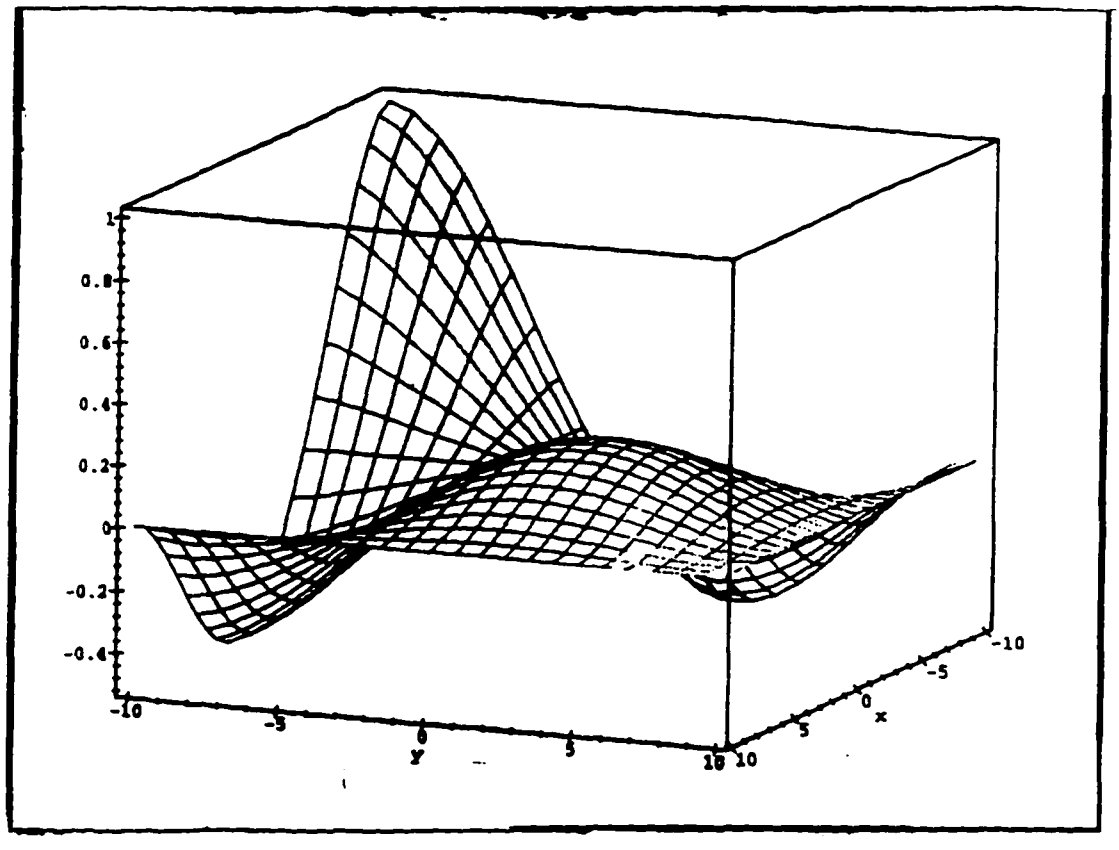

Figure 4.3: Error Due to the Corner Flux at (-a,-b)

4.b.1.c. Error Shape for Corner Current Error

By definition, the directed currents are proportional to one of the first order spatial derivatives of the two-dimensional flux evaluated at a point (the corners of a node for reconstruction purposes). For each corner, there will be a directed current in the x direction and a directed current in the y direction. The shapes for the directed corner currents are a bit deceptive since they do not appear to be rotations. Figures 4.4 and 4.5 present the errors due to the x directed corner current at (a,b) and at (-a,b), respectively.

The shapes of these two plots are similar with slightly different magnitudes. By examining the plots, it is difficult to conclude that these two surfaces have the same functional form.

However, by examining the representation for these error shapes:

$$-D_s \frac{\partial \phi(x,y)}{\partial x}\bigg|_{\substack{x=a \\ y=b}} \Rightarrow r_6 = \frac{a}{64D_s}\begin{pmatrix} -4\eta + 1 - 10\eta^4\xi + 8\eta\xi - 15\eta^4\xi^2 - 8\eta^3\xi - 6\eta^3 - 3\xi^2 \\ +4\eta^3 + 5\eta^4 + 12\eta^2\xi + 12\eta\xi^2 + 18\eta^2\xi^2 - 2\xi - 12\eta^3\xi^2 \end{pmatrix}$$

(Eq. 4.29)

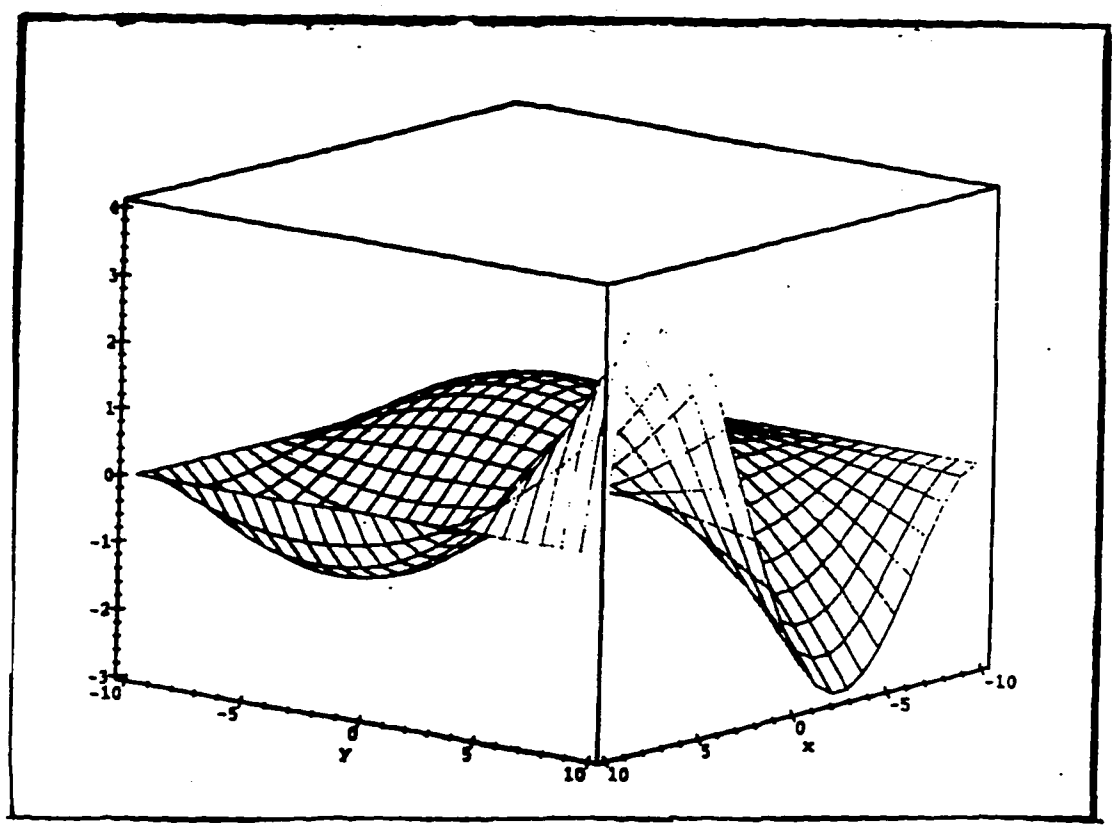

Figure 4.4: Error Due to the x-directed Corner Current at (a,b)

$$-D_c \left.\frac{\partial\, \phi(x,y)}{\partial\, x}\right|_{\substack{x=-a \\ y=b}} \Rightarrow r_c = \frac{a}{64 D_c}\begin{pmatrix} -12\eta^2\xi + 10\eta^4\xi + 4\eta^3 + 15\eta^4\xi^2 - 5\eta^4 - 4\eta + 8\eta\xi - 8\eta^3\xi \\ -12\eta^3\xi^2 + 2\xi + 3\xi^2 + 6\eta^2 + 12\eta\xi^2 - 1 - 18\eta^2\xi^2 \end{pmatrix}$$

(Eq. 4.30)

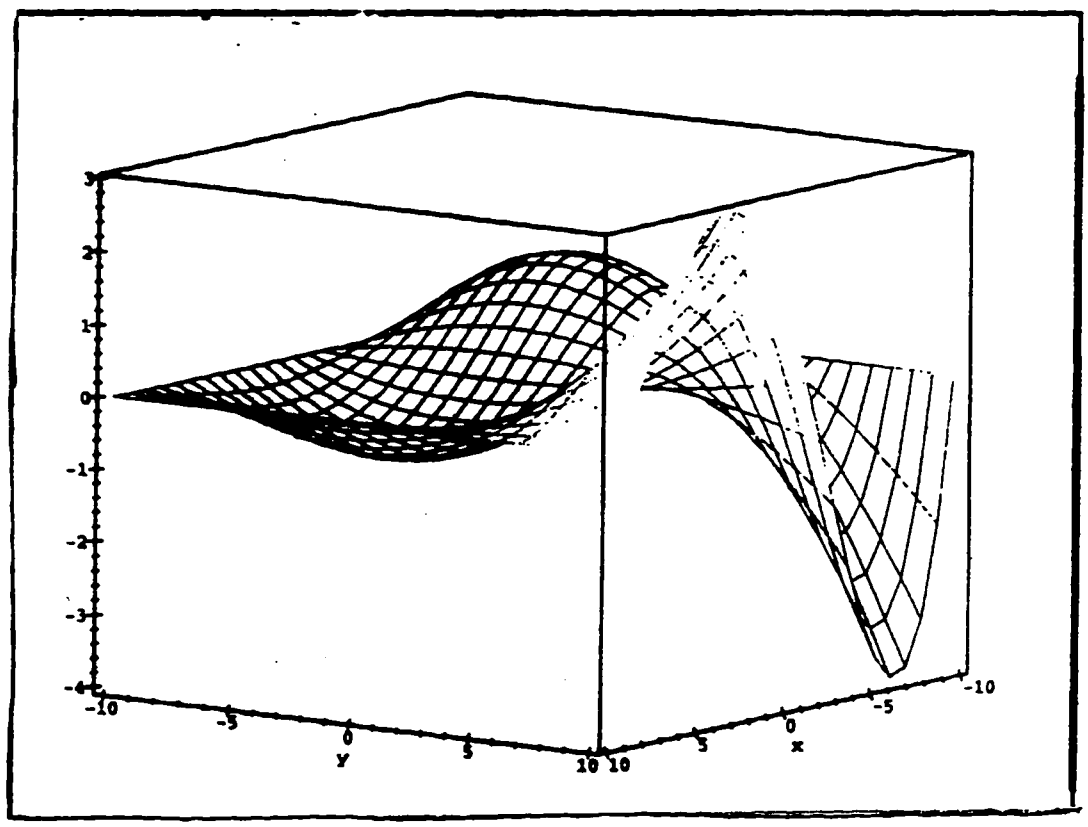

Figure 4.5: Error Due to x-directed Corner Current at (-a,b)

it is easy to show that Equation 4.30 is equivalent to Equation 4.29 when a is replaced by -a and η is replaced by $-\eta$. The functional forms for the remaining directed currents are

derived by placing the proper signs on the variables. The forms for the y directed currents are the same as the forms for the x directed currents except that the dependent variables are interchanged and the half width in the x direction (a) is replaced by the half width in the y direction (b).

With the corner fluxes, the error peaked at the corner and rapidly died away. Unfortunately with the x and y directed corner currents, the maxima and minima occur in approximately the same positions for related directed currents. By related directed currents, it is meant those directed currents that apply to the same face. For example, the x directed currents on the face y=b are located at (a,b), and (-a,b). These two directed currents are related and they have the same general locations for the peaks and the valleys. This can have both good and bad repercussions for the overall effect of the error. For example, if the two errors presented in Figures 4.4 and 4.5 had the same sign, then the total effect of the errors would add since they have approximately the same shape. Conversely, if the signs of the errors were opposite, then the errors would subtract and the overall effect on the error would diminish. This can have a significant effect on the total error. In the two figures, it can be seen that the maximum propagated effect of the error would be around four times the input error. If this effect was additive, then an error in the corner current would have effect about seven times as great as the input error.

Figures 4.2 and 4.5 represent the error due to the corner flux and the error due to the x-directed corner current at the point (-a,b). The error due to the flux is a maximum at this point while the error due to the x-directed corner current is zero at the same point. The absolute value of the error increases further away from the corner point. The fact

that the error in the x-directed current at the corner is zero should not be a cause for alarm. In fact, it is necessary that the x-directed current be zero at the corner point. The maxima of a function are determined by setting the gradient of the function equal to zero. Thus, a maxima of the corner flux error must correspond to zero error due to the directed corner currents.

Equations 4.29 and 4.30 reveal that the errors associated with the corner currents are dependent on the material properties (through the diffusion coefficient) and on the size of the node. For the shapes presented in Figures 4.4 and 4.5, a node size of 20 cm was used and the diffusion coefficient was taken from the thermal diffusion coefficient for plutonium. This is the smallest realistic diffusion coefficient that might be encountered in reactor analysis. The smallest diffusion coefficient is used because it is less than unity and it appears in the denominator. Thus, these results will be the most conservative since they will reveal the largest expected error.

The node size was taken from the IAEA2D benchmark problem. The larger the node size, the greater the effect of error propagation. In general, a twenty-centimeter wide node is commonly used in nodal analysis. This is approximately the size of an assembly in a PWR reactor core.

4.b.1.d. Error Shape for Surface Averaged Current Error

Continuing with the analysis of the error shapes, the largest errors occur with the surface averaged currents. These errors are large because they include the material properties of the node and the size of the node. In addition, they represent averaged values and are thus less accurate than a detailed value. Figure 4.6 is a plot of the error associated with the x directed current averaged over the y direction on the surface x=a. This will lead directly to a similar analysis concerning the relationship between the zeros of the directed currents and the maxima of the fluxes as was presented in the previous sections.

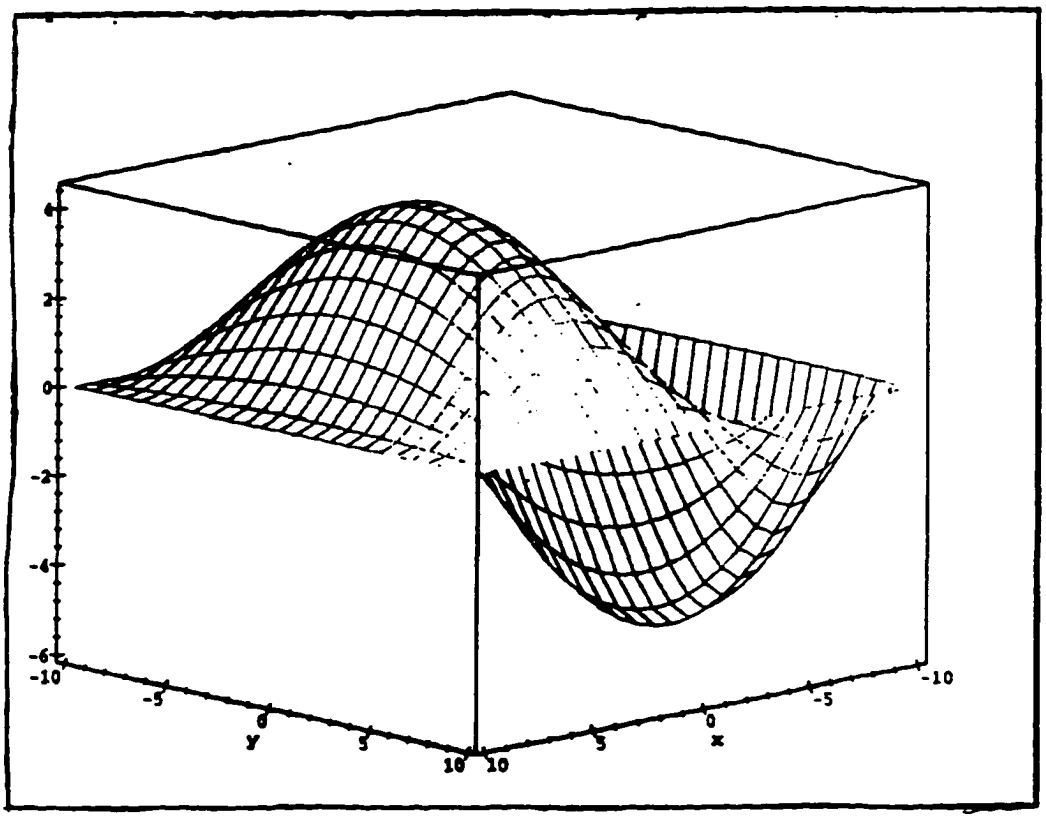

Figure 4.6: Error Due to x-directed, y-averaged Surface Current at x=a

$$\frac{-D_g}{2b}\int_{-b}^{b} dy \left.\frac{\partial \phi(x,y)}{\partial x}\right|_{x=a} \Rightarrow r_{14} = \frac{a}{64D_g}\begin{pmatrix} 24\eta - 6 + 36\eta^2 - 24\eta\xi^2 + 30\eta^4\xi^2 \\ -24\eta^3 - 36\eta^2\xi^2 + 24\eta^3\xi^2 + 6\xi^2 - 30\eta^4 \end{pmatrix}$$

(Eq. 4.31)

The plot for the opposite surface (x=-a) is identical to Figure 4.6 except that it is rotated 180° about the x=0 axis. Equivalently, this is found by replacing a with -a and η with $-\eta$. Again, with these averaged surface currents, the peaks and the valley appear in approximately the same locations so there is either a constructive increase in the error or

a cancellation of the error. The two plots for the y directed, x averaged surface currents are the same as the two plots presented above except that they are rotated 90° about the origin.

4.b.1.e. Error Shape for Surface Averaged Flux Error

The shapes for the surfaced averaged fluxes are approximately the same as the surface averaged currents except that the rotations are 180° about the origin for the two surface averaged fluxes in the same direction. This implies that there will be destructive composition when the signs of the errors are the same and constructive composition when the signs of the errors are opposite. The expression for the error associated with the x averaged surface flux on the surface y=-b is given by Equation 4.32.

$$\frac{1}{2a}\int_{-a}^{a} dx\phi(x,-b) \Rightarrow r_{21} = \frac{1}{64}\begin{pmatrix} 144\eta^2\xi + 90\eta^4\xi^2 - 90\xi^4 - 84\xi + 90\eta^2\xi^4 - 24\eta^2\xi^3 - 60\eta^4\xi \\ +198\xi^2 + 24\xi^3 - 48 + 78\eta^2 - 288\eta^2\xi^2 - 30\eta^4 \end{pmatrix}$$

(Eq. 4.32)

Figure 4.7 is the three-dimensional plot of this error. For the surface averaged fluxes, the peak error is located at the center of the surface. Again, the magnitude of the error is quickly reduced further away from the surface.

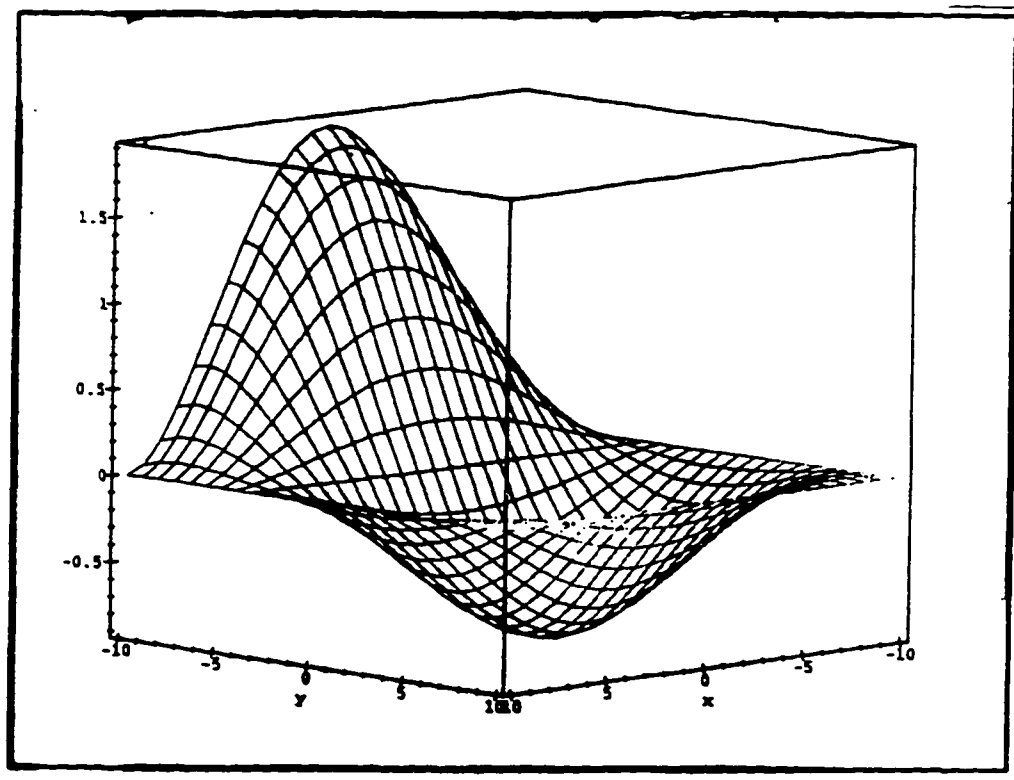

Figure 4.7: Error Due to the x-averaged Surface Flux at y=-b

4.b.2. Analysis of the Errors Associated with Nodal Input Data Error

With the exception of the node averaged flux, the peak in the error due to the nodal input data occurs at or near the edges of the node. For the currents with the same signs, these errors are additive while the errors cancel for the fluxes with the same sign. Further conclusions about how the errors relate to themselves is not possible since the magnitude and the sign of the input data error will not be known.

By integrating the error shapes over the dimensions of the node, an average effect of the input errors is determined. The average error for all of the input data is zero except

for the average error corresponding to the node average flux. The average error for the node average flux is one, which hindsight indicates is obvious. The average error in the reconstructed flux should be caused by the average error in the node average flux.

4.b.3. Determination of the Maximum Errors Due to Nodal Input Data Error

Perhaps more important than the average error is the maximum error associated with each piece of input data. Initially, it was assumed that the Karush-Kuhn-Tucker Theorem (gradient form)[21] would be used to determine the local maximum and minimum since these flux shapes have an analytic form. Even though the functions are convex (simple polynomials), it cannot be guaranteed that the maximum will not occur on the boundary of the node. As can be seen from the above figures, the maximum and minimum regularly appear on the surfaces of the node.

Simply setting the gradient of the error shape to zero will result in the global maximum and minimum locations. The local, in-node, maximum and minimum locations are required. To find these local maxima/minima, the gradient of the error shape is set to zero and the roots of the resulting equations are determined. Those locations outside the domain are discarded. Then the maxima and minima along each of the four surfaces are calculated. Finally, the value of error at the four corner points are determined. All of these values are compared to determine the maximum absolute value of the error in the node. Since there is a limited set of error shapes, those errors having the same functional form will have the same maximum absolute value located at a

different points. Table 4.5 lists the location and the magnitude of the error for each of the nodal inputs. For those shapes corresponding to currents, the maximum value of the shape is listed along with the value evaluated at a node half width of 10 cm and a diffusion coefficient of 0.34 cm. The errors associated with the currents have a value of a/D_g multiplying the tabulated error. By using these values for the material properties, the worst case is examined.

As can be seen in Table 4.5, the maximum values for the errors are in the same order of magnitude. With the exception of the node average flux, the maximum errors associated with the fluxes are approximately two to three times smaller than the errors corresponding to the derivatives of the flux (currents). It should be recalled that this analysis is for the worst case. With a larger value of the diffusion coefficient, the maximum values associated with the currents would approach the maximum values associated with the fluxes. This can be seen by multiplying the errors for the currents by the diffusion coefficient.

Even in the worst case, the maximum error for an individual piece of input data will be five times the input error. It should be restated that no conclusions can be made concerning the overall effect of the error in the input data. This is dependent on the sign and the magnitude of all of the individual components.

Input data	Location of maximum (x,y)	Maximum value of the error	Value of the error for D=0.34 and a=10
$\frac{1}{4ab}\int_{-a}^{a} dx \int_{-b}^{b} dy \phi(x,y)$	(0,0)	3.375	3.375
$\phi(a,b)$	(a,b)	1.0	1.0
$\phi(-a,b)$	(-a,b)	1.0	1.0
$\phi(-a,-b)$	(-a,-b)	1.0	1.0
$\phi(a,-b)$	(a,-b)	1.0	1.0
$-D_g \frac{\partial \sigma(x,y)}{\partial x}\Big\|_{x=a,y=b}$	(0.6899a,b)	0.1356	3.39
$-D_g \frac{\partial \sigma(x,y)}{\partial x}\Big\|_{x=-a,y=b}$	(-0.6899a,b)	-0.1356	-3.39
$-D_g \frac{\partial \sigma(x,y)}{\partial x}\Big\|_{x=-a,y=-b}$	(-0.6899a,-b)	-0.1356	-3.39
$-D_g \frac{\partial \sigma(x,y)}{\partial x}\Big\|_{x=a,y=-b}$	(0.6899a,-b)	0.1356	-3.39
$-D_g \frac{\partial \sigma(x,y)}{\partial y}\Big\|_{x=a,y=b}$	(a,0.6899b)	0.1356	3.39
$-D_g \frac{\partial \sigma(x,y)}{\partial y}\Big\|_{x=-a,y=b}$	(-a,0.6889b)	0.1356	3.39
$-D_g \frac{\partial \sigma(x,y)}{\partial y}\Big\|_{x=-a,y=-b}$	(-a,-0.6889b)	-0.1356	-3.39
$-D_g \frac{\partial \sigma(x,y)}{\partial y}\Big\|_{x=a,y=-b}$	(a,-0.6889b)	-0.1356	-3.39

Table 4.5: Locations and Maximum Values of the Errors for the Nodal Input Data

(part I)

Input data	Location of maximum (x,y)	Maximum value of the error	Value of the error for D=0.34 and a=10
$\frac{-D_e}{2b}\int_{-b}^{b} dy \frac{\partial\ \phi(x,y)}{\partial x}\Big\|_{x=a}$	(0.6899a,0)	0.2034	5.085
$\frac{-D_e}{2b}\int_{-b}^{b} dy \frac{\partial\ \phi(x,y)}{\partial x}\Big\|_{x=-a}$	(-0.6899a,0)	-0.2034	-5.085
$\frac{-D_e}{2a}\int_{-a}^{a} dx \frac{\partial\ \phi(x,y)}{\partial y}\Big\|_{y=b}$	(0,0.6889b)	0.2034	5.085
$\frac{-D_e}{2a}\int_{-a}^{a} dx \frac{\partial\ \phi(x,y)}{\partial y}\Big\|_{y=-b}$	(0,-0.6889b)	-0.2034	-5.085
$\frac{1}{2b}\int_{-b}^{b} dy\phi(a,y)$	(a,0)	1.875	1.875
$\frac{1}{2b}\int_{-b}^{b} dy\phi(-a,y)$	(-a,0)	1.875	1.875
$\frac{1}{2a}\int_{-a}^{a} dx\phi(x,b)$	(0,b)	1.875	1.875
$\frac{1}{2a}\int_{-a}^{a} dx\phi(x,-b)$	(0,-b)	1.875	1.875

Table 4.5: Locations and Maximum Values of the Errors for the Nodal Input Data

(part II)

4.c. Figure of Merit

Now that both types of error have been quantified, it is necessary to determine a method that can be used to evaluate the 'goodness' of a flux reconstruction scheme. This method should include the total deviation due to the method plus give an indication as to how sensitive the method is to the errors in the input data. A first guess would be to add the total deviation from the polynomial and hyperbolic case to the maximum error from the nodal input data. This method has a major weakness in that the value of the total deviation dwarfs the value of the maximum error from the nodal input data.

A more equitable approach would be to multiply the total deviation by the maximum error. The resulting number would not really have any physical meaning but it provides a reference to the 'goodness' of the flux reconstruction method. For the case examined, this figure of merit would be: F=5666.7. A lower value for the figure of merit would imply either that the maximum nodal input error was smaller or that the total deviation was smaller. Either of these two results would imply a more accurate flux reconstruction scheme and, thus, an improvement over the original method. The objective in improving the flux reconstruction methodology would be to find the lowest practical figure of merit.

5. Attempts at Reducing the Errors of Nodal Flux Reconstruction

In the preceding chapter, a method was developed to characterize the "goodness" of a nodal reconstruction scheme. This chapter will investigate a number of methods that may reduce the Figure of Merit for a nodal reconstruction. This reduction in the Figure of Merit directly implies an improvement in the nodal reconstruction scheme. The attempts at reducing the nodal reconstruction error can be grouped into three major categories: 1) those methods that alter the basis functions; 2) those methods that alter the set of functionals used; and 3) new models. The latter category could also be considered as a combination of the first two categories.

5.a. Variation of the Basis Functions

An obvious approach to reducing the nodal reconstruction errors would be to determine an optimal set of basis functions used to represent the two-dimensional flux. The original work by Wagner and Koebke used simple polynomials. Later, the use of hyperbolic basis functions revealed more accurate reconstructed fluxes for some test cases. In keeping with the theme of using "engineering" knowledge to restrict the scope of this investigation, various sets of basis functions were investigated that satisfy the governing two-dimensional differential equations.

5.a.1. Orthogonal Polynomials

As a continuation of the original work by Wagner and Koebke, various sets of orthogonal polynomials were chosen as basis functions for the reconstructed flux. Since the region is symmetric, Hermite and Legendre polynomials were investigated as possible improvements for the reconstruction method. Surprisingly, using these sets of orthogonal polynomials as basis functions produces the exact same two-dimensional, reconstructed flux as the simple polynomial basis functions. Thus, there is no improvement in the nodal reconstruction method.

5.a.2. Hyperbolic Functions

As was mentioned earlier, hyperbolic functions will satisfy the governing, two-dimensional differential equations. However, their choice as a set of basis functions provides additional complications. There must be a constant in the argument of the hyperbolic functions. In the literature, this constant has generally been assumed to be dependent on the material properties of the node. As a first attempt, this work will not make any assumptions concerning this constant except that this parameter will indeed be constant and that it will be directionally independent inside the node.

Unfortunately, using a completely general parameter makes the problem impossible to solve symbolically. The constraint matrix is too dense to be symbolically inverted before a computational time limit is reached. A series of approximations were

then attempted to make the problem tractable: (1) square node (a=b), (2) specifying the size of the node (a=b=10 cm), and (3) selecting specific values for the hyperbolic parameter. It was only after applying all three approximations that the software was able to invert the constraint matrix. These approximations reduced the problem to a numerical problem and not a symbolic problem. This would allow the calculation of the functional interpolation error for a series of chosen material properties, thus making the analysis dependent upon the numeric benchmark chosen. This dependence upon the numeric benchmark would be acceptable for a given reactor model since the material properties and physical dimensions would be fixed. However, this dependence defeats the objective of determining a method by which any form of two-dimensional flux reconstruction methods could be compared to determine the optimal method.

The numerical results were disappointing. If the proper basis functions were 'guessed' correctly, then the functional interpolation procedure produced decent results. But, if the choice of basis functions were incorrect, then serious errors occurred. In addition, the magnitude of the propagated error due to the nodal input data was directly related to the choice of material properties. Unfortunately, the responses of the reconstructed fluxes were not consistent; what approximated the hyperbolic sine well with small material properties did not approximate the hyperbolic sine well with large material properties.

As an example of lack of consistency for hyperbolic basis functions, the set of hyperbolic basis functions chosen by Koebke and Hetzelt[17] were investigated. The expression for the two-dimensional flux was given by:

$$\Phi_g(x,y) = \sum_{i,j=0}^{4} c_{ij} F_i(x) F_j(y) \qquad \text{Eq. 5.1}$$

where:

$$F_0(v) = 1 \qquad F_1(v) = \sinh(kv) \qquad F_2(v) = \cosh(kv)$$

$$F_3(v) = \sinh(2kv) \qquad F_4(v) = \cosh(2kv)$$

with v representing either x or y.

Three different values for k were examined with a node half width of 10 cm. The results for the total deviations and the average errors for five combinations of hyperbolic functions are presented in Table 5.1 and Table 5.2.

	k=0.1414	k=0.4472	k=0.5701
cosh(kx)	-0.11×10^{-6}	-85.9678	26.2377
sinh(kx)	-1.36818	-9.78535	0.13×10^{-6}
cosh(kx)sinh(ky)	0.10×10^{-6}	0.00000	-26.2377
cosh(kx)cosh(ky)	-0.32×10^{-6}	95.7531	688.4206
sinh(kx)sinh(ky)	-1.8719	-0.86×10^{-7}	-688.4206

Table 5.1: Average Error for Three Values of k

	k=0.1414	k=0.4472	k=0.5701
cosh(kx)	35.1783	967.308	4.6685
sinh(kx)	68.1951	3.11956	4.6281
cosh(kx)sinh(ky)	10.5280	1.12682	0.9985
cosh(kx)cosh(ky)	48.4305	2.61766	3.2812
sinh(kx)sinh(ky)	200.9848	50.6397	6.2787

Table 5.2: Total Deviations for Three Values of k

As can be seen from Table 5.1, there is no consistency in the behavior of the values of the average error. The same definition of average error is used here as was used in Chapter 4; the average error is the integral over the entire node of the difference between the true flux and the reconstructed flux.

The average error of the hyperbolic sine is small when k is small and then increases slightly as k increases. However, when k becomes large, the average error of the hyperbolic sine approaches zero. Conversely, the average error of the product of a hyperbolic sine and hyperbolic cosine is approximately zero until k becomes large. At this point, the average error begins to grow rapidly.

Similar inconsistencies appear in the calculation of the total deviations (presented in Table 5.2). In this table, the value of the total deviation for the hyperbolic cosine begins with a moderate value and then increases rapidly as k increases. This increase continues until k becomes too large. At this point, the total deviation drops to a much

until k becomes too large. At this point, the total deviation drops to a much smaller value. This behavior is not repeated in any other combination of hyperbolic functions.

An analysis of the nodal input errors for this form of reconstruction reveal that the input errors will exhibit a decrease in magnitude as the value of the constant increases. This decrease in magnitude can be seen in Figures 5.1 and 5.2.

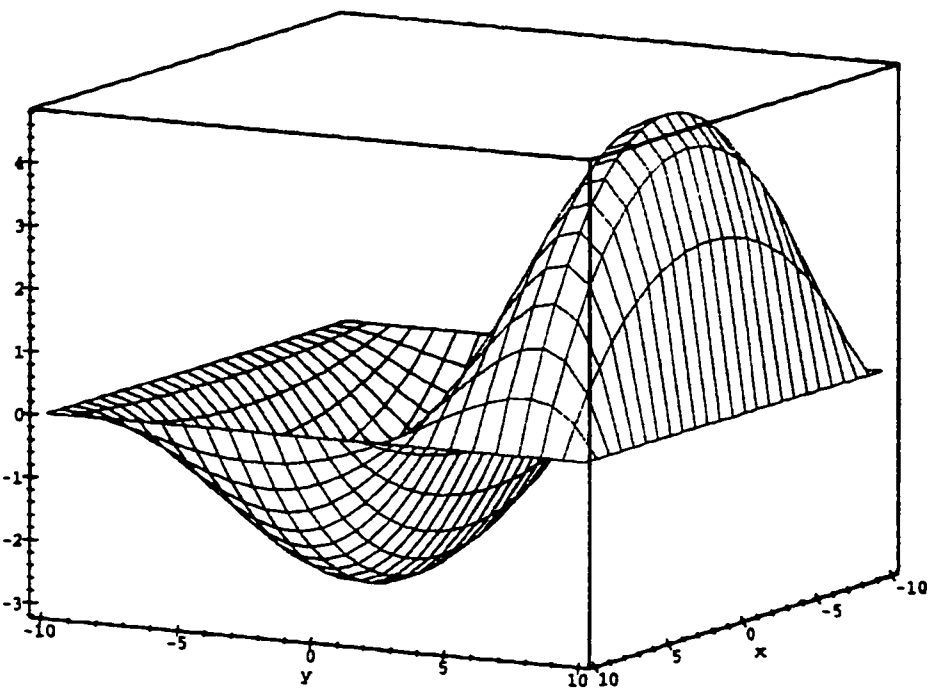

Figure 5.1: Error Shape Due to the y-directed, x-averaged Surface Current at y=b for k=0.1414

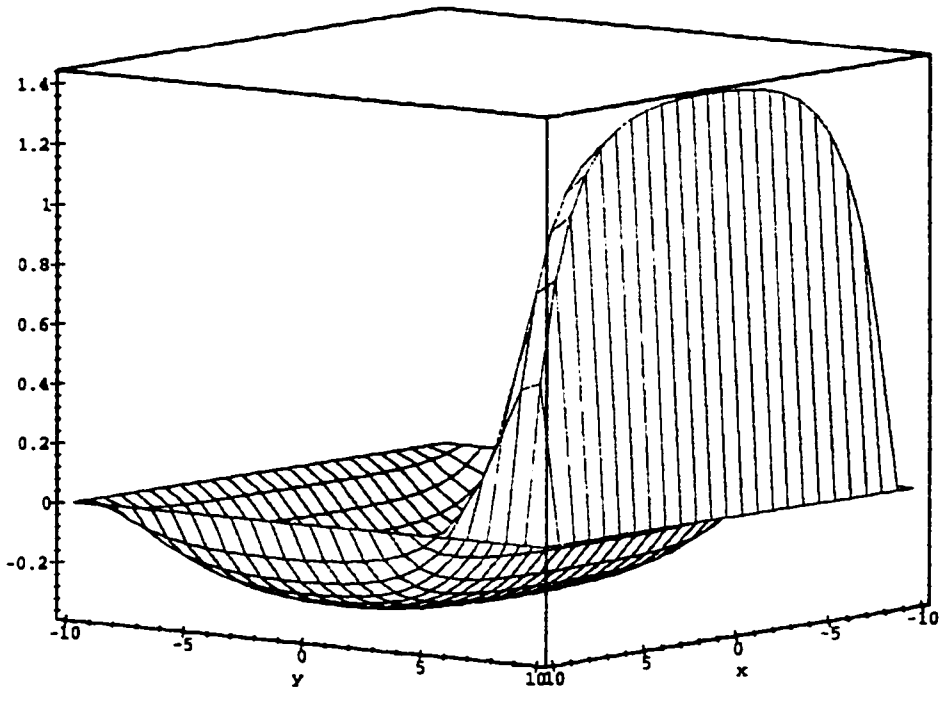

Figure 5.2: Error Shape Due to the y-directed, x-averaged Surface Current at y=b for k=0.5701

The figures presented here represent the most extreme example of the diminishing of the nodal input error with increasing k. The maximum value of the error due to this piece of nodal input data is decreased by more than half of it's initial value. For other pieces of input data, this reduction in error is not as dramatic.

One of the goals of this work was to find a method to compare various nodal flux reconstruction methods. The inconsistencies in the evaluation of the error due to the functional interpolation method and the application of the three approximations have

restricted the problem to the extent that comparison between a hyperbolic reconstruction method and a polynomial reconstruction method would produce no tangible results.

5.b. Variation of the Set of Functionals

It is possible that the original functional set chosen by Wagner and Koebke was not the optimal choice of functionals. An investigation of other sets of functionals, using the original polynomial basis functions, may reveal if there exists a superior set. The original polynomial basis functions are used for comparative purposes in order to isolate the causes of any improvements. For now, the set of functionals will be required to have twenty-one elements.

If the two-dimensional flux expansion had more terms, then it might be possible that this expanded representation would be more accurate. Instead of 'inventing' new functionals, it was decided to investigate the possibility of using the governing differential equations to produce additional terms for the flux expansion. This method revolves around the idea that a flux expansion containing more than twenty-one terms could be inserted into the governing differential equations. After performing the differentiation of the flux expansion, it might be possible to write new coefficients in terms of the original twenty-one coefficients. In this way, one would perform traditional flux reconstruction and then use this new information to generate additional constants. Speaking precisely, this method would not increase or change the set of functionals but it would increase the order of the approximation to the two-dimensional flux.

Beginning with the two group diffusion equation written in a general form:

$$\nabla^2\Phi_1(x,y)+a_1\Phi_1(x,y)+a_2\Phi_2(x,y)=0$$
$$\nabla^2\Phi_2(x,y)+b_1\Phi_1(x,y)+b_2\Phi_2(x,y)=0 \qquad \text{Eq. 5.2.}$$

For the moment, the exact representations of the constants a_i and b_i are not important. The first attempt to use Equations 5.2 to produce a more accurate flux reconstruction consisted of inserting assumed forms for the fast flux ($\Phi_1(x,y)$) and for the thermal flux ($\Phi_2(x,y)$). After performing the perquisite derivatives, various relationships between unknown coefficients for the fast and the thermal flux were determined. At this point, it was realized that one could write some of the coefficients of the thermal flux in terms of the coefficients of the fast flux. However, the error analysis that was presented in Chapter 4 can only analyze a mono-energetic, two-dimensional flux representation. Thus, this avenue of investigation was not pursued further.

Fortunately, the fast flux may be isolated in the second differential equation presented in Equation 5.2:

$$\Phi_1(x,y)=\frac{-1}{b_1}\left[\nabla^2\Phi_2(x,y)+b_2\Phi_2(x,y)\right]. \qquad \text{Eq. 5.3}$$

Inserting this form for the fast flux into the first differential equation presented in Equation 5.2 will result in one differential equation that depends solely on the thermal flux (after some manipulation):

$$\nabla^2\left(\nabla^2\Phi_2(x,y)\right)+d_1\nabla^2\Phi_2(x,y)+d_2\Phi_2(x,y)=0. \quad \text{Eq. 5.4}$$

where the two constants, d_1 and d_2, are written in terms of the original four constants. A bi-variate polynomial form of the reconstructed flux with twenty-five unknown coefficients is then chosen:

$$\Phi_2(x,y)=\sum_{i=0}^{4}\sum_{j=0}^{4}c_{ij}x^iy^j. \quad \text{Eq. 5.5}$$

Equation 5.5 could have been extended to any order but it was written in this form in order to have twenty-five coefficients instead of the twenty-one coefficients that appear in the traditional flux reconstruction method. This form for the reconstructed flux is then inserted into Equation 5.4 resulting in:

$$\sum_{i=0}^{4}\sum_{j=0}^{4}\left\{\begin{array}{l}c_{ij}(i)(i-1)(i-2)(i-3)x^{i-4}y^j+c_{ij}(j)(j-1)(j-2)(j-3)x^iy^{j-4}\\+2c_{ij}(i)(i-1)(j)(j-1)x^{i-2}y^{j-2}+d_1c_{ij}(i)(i-1)x^{i-2}y^j\\+d_1c_{ij}(j)(j-1)x^iy^{j-2}+d_2c_{ij}x^iy^j\end{array}\right\}=0. \quad \text{Eq. 5.6}$$

By comparing like powers of $x^i y^j$ and performing some extensive algebra, it was determined that four new, unknown coefficients that were introduced in the new flux expansion could not be written in terms of the original coefficients. In fact, this analysis leads to the conclusion that ALL of the coefficients of the flux expansion should be equal to zero.

5.c. New Models for Reconstructing the Flux

Some attention was given to devising methods not based on the traditional form of flux reconstruction. One method involved the examination of the definition of a transverse integrated flux:

$$\overline{\Phi}_g(x) = \frac{1}{2b} \int_{y=-b}^{b} dy \Phi_g(x,y) . \quad \text{Eq. 5.7}$$

This relates the transverse integrated flux to the two-dimensional flux. The transverse integrated flux is also found from the results of a nodal code; the exact form of the flux depends on the method of solution of the nodal code. For example, ILLICO[6] produces the following form for the transverse integrated flux:

$$\overline{\Phi}_g(x) = E\sinh(kx) + F\cosh(kx) + \sum_{i=0}^{2} a_i P_i(x/a) . \quad \text{Eq. 5.8}$$

The actual form for the reconstructed flux does not matter for these purposes. These two representations for the reconstructed flux, Equations 5.7 and 5.8, are now equated:

$$E\sinh(kx)+F\cosh(kx)+\sum_{i=0}^{2}a_iP_i(x/a)=\frac{1}{2b}\int_{y=-b}^{b}dy\Phi_g(x,y). \quad \text{Eq. 5.9}$$

Equation 5.9 relates the two-dimensional flux to the results from a nodal code. However, there is still a directional dependence in existence in Equation 5.9. If Equation 5.9 is operated upon by a series of weight functions and then integrated over the entire remaining direction, then this resulting equation would be cast in the form of a functional interpolation problem:

$$\int_{x=-a}^{a}dxG_l(x)\left(E\sinh(kx)+F\cosh(kx)+\sum_{i=0}^{2}a_iP_i(x/a)\right)=\int_{x=-a}^{a}dxG_l(x)\left(\frac{1}{2b}\int_{y=-b}^{b}dy\Phi_g(x,y)\right). \quad \text{Eq. 5.10}$$

To solve Equation 5.10, one chooses a set of basis functions for the two-dimensional flux and then selects enough weighting functions ($G_l(x)$) in order to have enough constraints. Care must be taken to choose weighting functions in both directions and to select the weighting functions so that the resulting functionals are linearly independent in the algebraic conjugate space.

Despite extensive searching for proper combinations of weighting functions and flux representations, the model never produced a constraint matrix that was not singular.

Great care was taken to ensure that all of the mathematical formalities concerning functional interpolation were followed: the functionals were independent in the algebraic conjugate space and the chosen flux representations were linearly independent. Ultimately, this model was deemed a failure because it was unable to produce a non-singular constraint matrix.

5.d. Conclusions

Unfortunately, varying the set of functionals either produced a reconstructed flux that was not mono-energetic or it determined that all of the coefficients of the reconstructed flux should equal zero.. This is an area that needs further investigation since there might exist other combinations of basis functions or other sets of functionals that would minimize the total error associated with the reconstructed flux. Additional work needs to be done in the field of developing new models for nodal flux reconstruction. Without advances in all of these fields, it will be impossible to determine a method for indicating which nodal flux reconstruction scheme is optimal.

6 New Applications of Nodal Flux Reconstruction

In this chapter new methods for applying the reconstructed flux will be presented. In addition, other applications of the methods developed in this thesis (both the error analysis and the functional interpolation) will be investigated.

6.a. Validation of the Diffusion Approximation

Another application of the reconstructed flux could be the determination of the applicability of the diffusion approximation. It should be remembered that the diffusion equation is an approximation to the transport equation. Under certain conditions, the validity of this approximation breaks down. One of these limiting conditions is that the two dimensional flux does not vary 'too rapidly.'

The derivative can be considered as a measure of how rapidly the value of a function is changing. A large absolute value of the derivative indicates that the value of the function is changing rapidly. Since the reconstructed flux is a good representation of the two dimensional flux, it may be used to determine if the function is changing too rapidly. However, because this is a two dimensional function, the following will be used to estimate if the function is varying too rapidly:

$$\omega^2 = \left(\frac{\partial \phi_{g,\mathrm{RECON}}(x,y)}{\partial x}\right)^2 + \left(\frac{\partial \phi_{g,\mathrm{RECON}}(x,y)}{\partial y}\right)^2 \qquad \text{(Eq. 6.1)}$$

The value of ω may be analytically calculated from the reconstructed flux and then compared to a pre-selected maximum value of ω_{max}. Those nodes that have values of ω greater than the preset maximum value should be recalculated with a transport formulation since the diffusion approximation breaks down in these nodes.

Unfortunately, this approach only works well in theory. Simply calculating the first derivatives results in seven pages of equations (see Appendix 2 for a printout of a MAPLE session that calculated this derivative). Squaring these terms, adding them and then taking the square root of the resulting expression results in an expression approximately 200 pages long. It is unfeasible to encode this type of expression into a neutronics code, at least in a Fortran code. It might be possible to create a neutronics code that would operate in the environment of a symbolic manipulator in which case this approach may be more feasible.

6.b. Application of Functional Interpolation to Monte Carlo Results

It is desired to apply the methodology developed in this thesis to problems not directly related to nodal methods. In order to use this method, data that is functionally dependent on the global or heterogeneous flux is required. This functional dependence can be in the form of an integral average, a local derivative, or a pointwise evaluation of the flux to name a few possibilities. Theoretically, this method to determine the two-dimensional flux could be applied to numerical methods that are used in neutronics and other engineering applications.

6.b.1. Monte Carlo Methods

The Monte Carlo method discretizes the volume of interest into semi-homogeneous regions and then tracks individual particles (neutrons) to determine their energy, their location, and the type of interaction that they undergo. A typical Monte Carlo code tallies this information and interprets these results to determine the energy dependent fluxes in specified volumes or regions. Since this type of code analyzes individual neutron histories, the output from a Monte Carlo code is necessarily statistical in nature. This means that a flux result for a given region will be the statistical average flux in that region with an associated standard deviation. This standard deviation is related to the inverse of the square root of the number of histories. The code may be run with a large number of histories, thus reducing the error associated with the results.

Recently, the University of Illinois Reactor Laboratory purchased the Research Reactor Analysis Program (RRAP) from Atoms Analysis.[22] This code is a graphics package that interprets the results of the Monte Carlo N-Particle (MCNP) code that was developed at Los Alamos National Laboratory.[23] The RRAP code has been adapted to the configuration of the TRIGA reactor. The configuration is a hexagonal lattice with cylindrical fuel elements in the center of each hexagon. Figure 6.1 depicts the entire TRIGA core configuration.

The prototype version of the code was installed and benchmarked against experimental values determined at the beginning of the core lifetime in 1969. There was relatively good agreement between the results of the code and the experimental results except for the peak fluxes in the core. However, the RRAP code predicted peak fluxes that were in excess of twice the 1969 recorded values.

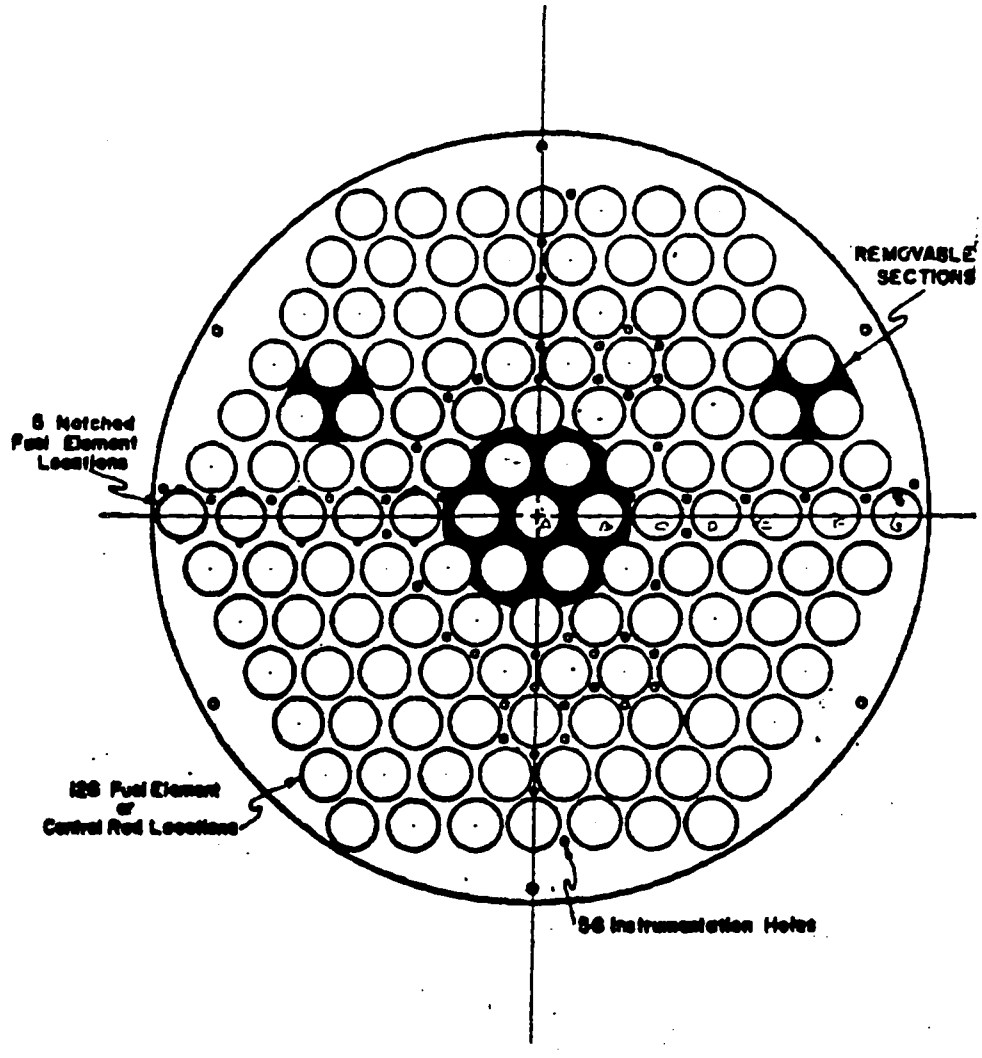

Figure 6.1: TRIGA Reactor Core Configuration[24]

Since this was a prototype version of the code, it was initially assumed that there might exist a 'bug' either in the input reactor configuration or in the material properties

(mainly in the temperature effects and the possible presence of Erbium in the fuel). However, careful examination revealed that the material properties were properly represented and that the configuration was correct. This difference in the peak fluxes could not be explained from a computational point of view. The records from 1969 are sketchy but they do indicate that the fluxes were determined from reliable methods (gold foil activation) and that these results were consistent with current peak flux measurements. Initially, it was assumed that a sample would be located in the region of peak flux (the central thimble) but no facility records could be found to computationally verify this assumption.[25] After much consideration, it was concluded that the measured peak flux and the calculated peak flux might not be in the same physical location.

6.b.2. Location of the Peak Flux

The peak flux should occur in the central region of the hexagonal array. This position, known as the central thimble, contains an irradiation tube that was vertically positioned in the core and then fixed in that position. This allowed for reproducibility of the fluxes. The axial positioning in the core is not believed to have been changed since 1969.[25] It is postulated that the sample used to calculate the peak flux could be above or below the location of the true peak axial flux. This true peak in the axial flux should occur slightly below the axial mid-plane since the control rods are removed from the top.

there will be a lower rate of neutron absorption at the bottom of the core and a corresponding higher flux towards the bottom of the core.

In the RRAP code, each fuel element and each irradiation chamber is treated individually in the radial direction. The axial portion of each position is divided into five equal sections three inches in length as is shown in Figure 6.2. The RRAP code provides the total flux and the thermal flux for each of the five axial divisions.

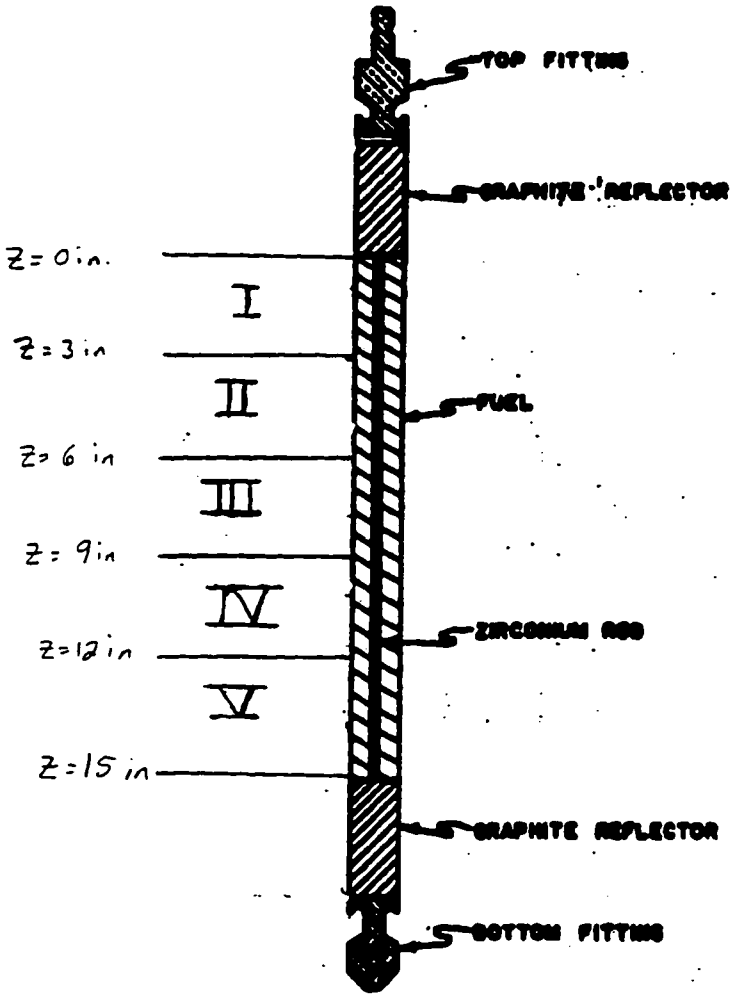

Figure 6.2: Axial Cross Section of A TRIGA Fuel Element[24]

For the initial core configuration, Table 6.1 tabulates the results obtained for each of the axial regions in the central thimble:

Region	Axial Position (in)	Midpoint (z=0 at top)	Total Flux n/cm^2 sec	Thermal Flux n/cm^2 sec
I	$0 \leq z \leq 3$	1.5 in	2.48048×10^{13}	1.81876×10^{13}
II	$3 \leq z \leq 6$	4.5 in	6.45794×10^{13}	4.34976×10^{13}
III	$6 \leq z \leq 9$	7.5 in	10.21210×10^{13}	6.88225×10^{13}
IV	$9 \leq z \leq 12$	10.5 in	9.39515×10^{13}	6.47688×10^{13}
V	$12 \leq z \leq 15$	13.5 in	4.38444×10^{13}	3.28208×10^{13}

Table 6.1: Axial Flux Values from the Monte Carlo Code

The numbers that were experimentally determined from gold foil activation are a total flux of 4.40×10^{13} n/cm^2 sec and a thermal flux a 3.02×10^{13} n/cm^2 sec.[25]

It is assumed that the fluxes from the RRAP code are accurate in order to determine the actual location of the sample used in the 1969 flux measurements. An axial function is constructed to fit the RRAP data. This reconstructed axial flux is written as a function of axial position. The experimentally determined flux provides a number. The reconstructed flux is then equated to this number and the axial position that corresponds to this flux value is determined. This procedure is repeated for the fast and the thermal flux to check if the resulting axial positions are consistent. It will be determined if the gold foil activated sample was located in the position of the highest flux in the core from the calculated thermal and fast flux distributions. The calculated

positions are compared to the locations of the maximum flux and these locations of maximum flux are determined by examining the axial profiles generated from the RRAP data.

6.b.2.a. Location by Linear Interpolation

The simplest method to determine the location of the sample would be to linearly interpolate between the two fluxes that bracket the experimental value. However, a problem arises since the MCNP results are not pointwise results but averages over a region. A first approximation assumes that the flux in a region can be represented by the MCNP value for that region located in the axial center of the region. The location of the axial center for each region was given in Table 6.1.

Since the flux is approximately parabolic in shape, there will be two locations that could bracket the experimental flux. It is assumed that the flux is approximately parabolic in shape due to the axial configuration of the core. In the axial direction, the core may be considered as a cylinder with reflectors on the top and the bottom of the fuel. This configuration leads to a form for the flux that is either a parabola or a shifted cosine.[26]

By performing linear fits for both the total flux and the thermal flux, the calculated axial locations that most closely agree will be taken as the location of the sample. The general form for linear interpolation is given by Equation 6.1:

$$\frac{\Phi(z_2)-\Phi(z)}{\Phi(z_2)-\Phi(z_1)}=\frac{z_2-z}{z_2-z_1}. \quad \text{(Eq. 6.1)}$$

To find the location of the total flux, $\Phi(z)$ is assumed to be the experimental value and $\Phi(z_1)$ and $\Phi(z_2)$ are taken from the RRAP results that bracket the experimental value. The values of z_1 and z_2 are axial positions corresponding to the two total flux values that bracket the experimental value. Equation 6.1 is then used to interpolate the location of the gold foil:

$$\frac{6.45794\times10^{13}-4.40\times10^{13}}{6.45794\times10^{13}-2.48048\times10^{13}}=\frac{4.5-z}{4.5-1.5}. \quad \text{(Eq. 6.2)}$$

Thus, the theoretically derived estimate for the upper location of the total flux is 2.95 inches from the top of the central thimble.

Since the flux is assumed to be parabolic (cosine) in shape, there will exist two axial locations in the core that may bracket the experimental value: one in the upper half of the core and one in the lower half of the core. In Table 6.2, it is seen that there is much greater agreement between the calculated location in the upper half of the core. In addition, it should be pointed out that the results for the lower half of the core require extrapolation and not interpolation since the flux values are less than the MCNP fluxes for region V. From these linear interpolation results, it is concluded that the 1969 measurements were taken at a location of approximately 2.95 inches from the top of the core not at the center of the core (7.5 inches from the top of the core).

Region	Location of total flux	Location of thermal flux
Upper half	2.95 in	2.92 in
Lower half	13.51 in	13.75 in

Table 6.2: Locations of Peak Flux from Linear Interpolation

6.b.2.b. Location by Polynomial Interpolation

Unfortunately, linear interpolation is not the most accurate approximation technique. The next approximation would be to attempt polynomial interpolation for the five regions. Polynomial interpolation also requires pointwise data points since polynomial interpolation selects a form for the flux and then constrains this form for the flux to fit the true flux at specific point (see Chapter 3). The same approximation will be utilized here as was used in the linear interpolation. The five data points, evaluated at the center of each region, will admit a fourth order polynomial function to approximate the axial flux:

$$\phi(z) = a_0 + a_1 z + a_2 z^2 + a_3 z^3 + a_4 z^4. \quad \text{(Eq. 6.3)}$$

This approximation is evaluated at each of the five data points thereby establishing a set of five constraint equations for the five unknowns (a_0, a_1, a_2, a_3, and a_4). Solving this set of linear equations for both the total and the thermal fluxes results in the coefficients given in Table 6.3.

Flux	a_0	a_1	a_2	a_3	a_4
Total	3.0587×10^{13}	-1.4324×10^{13}	0.8202×10^{13}	-0.0852×10^{13}	0.0024×10^{13}
Thermal	2.3167×10^{13}	-1.0393×10^{13}	0.5524×10^{13}	-0.0563×10^{13}	0.0016×10^{13}

Table 6.3: Coefficients for Polynomial Representation of the Fluxes

The axial location of the sample is found by setting the flux representation to the appropriate measured result. Using the fourth order polynomial flux representation and the above coefficients, the locations of the maxima and their values are computed. They are shown in Table 6.4. These maximum fluxes are obtained by differentiating the flux representation with respect to z and then equating this derivative to zero. The roots of this resulting equation are then the locations of the global maxima and minima. The required local maximum is found by eliminating those maxima that are located outside the domain of interest (that is, those maxima/minima that do not fall in the range of $0 \leq z \leq 15$ in).

Flux	Sample location	Maximum location	Maximum flux value
Total	3.24 in	8.54 in	10.527×10^{13} n/cm^2 s
Thermal	3.25 in	8.65 in	7.134×10^{13} n/cm^2 s

Table 6.4: Calculated Locations in the Core Based on Polynomial Interpolation

Again, there is very good agreement between the two calculated locations except that the results from polynomial interpolation indicate that the sample was about one half inch further in the core than the results from linear interpolation. From these calculations, it is evident that the maximum currently available total flux in the reactor is less than forty two percent of the calculated maximum total flux. This implies that if the central thimble irradiation tube could be lowed further into the core, then the total flux would be increased by a factor of 2.4 times the current maximum flux.

6.b.2.c. Location by Lagrange Interpolation

Another form of polynomial interpolation is Lagrange interpolation. In this method, the flux is represented by:

$$\phi(z) = a_0\ell_0(z) + a_1\ell_1(z) + a_2\ell_2(z) + a_3\ell_3(z) + a_4\ell_4(z). \quad \text{(Eq. 6.4)}$$

where

$$\ell_k(z) = \prod_{\substack{i=0 \\ i \neq k}}^{n} \frac{z - z_i}{z_k - z_i} \quad k = 0,1,\ldots,n \qquad \text{(Eq. 6.5)}$$

The ℓ's are known as Lagrange polynomials. It should be noted that the results from the polynomial interpolation and the results from Lagrange interpolation should be identical. There exists only one unique polynomial that will interpolate a discrete set of data.[1] In fact, the interpolation using Lagrange polynomials does reproduce the same results as were found by polynomial interpolation. Figures 6.3 and 6.4 present the plots of the total and the thermal flux from polynomial interpolation.

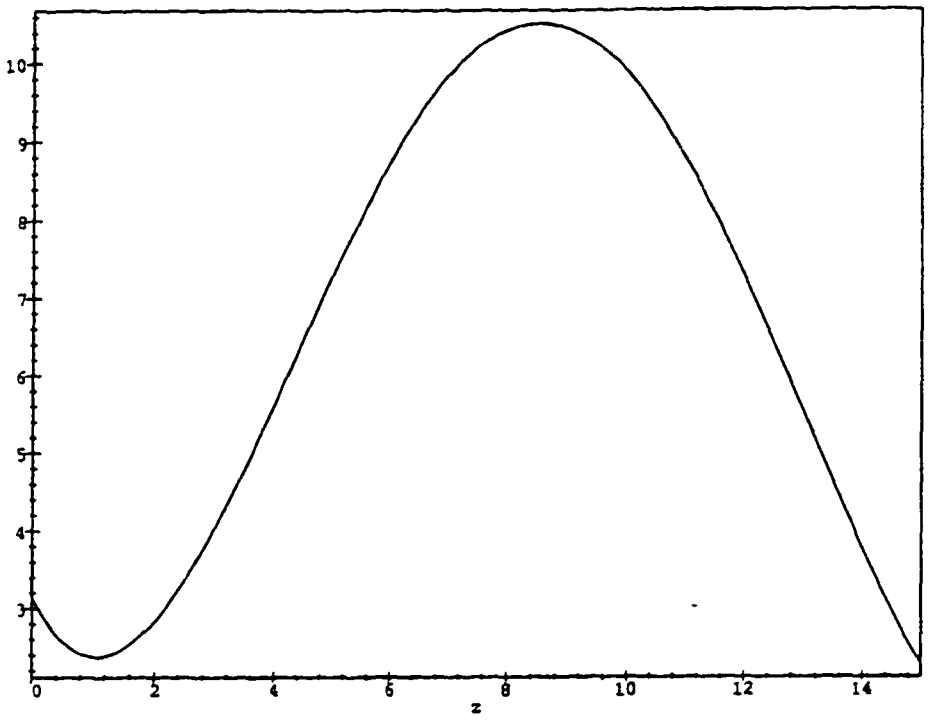

Figure 6.3: Total Flux from Polynomial Interpolation

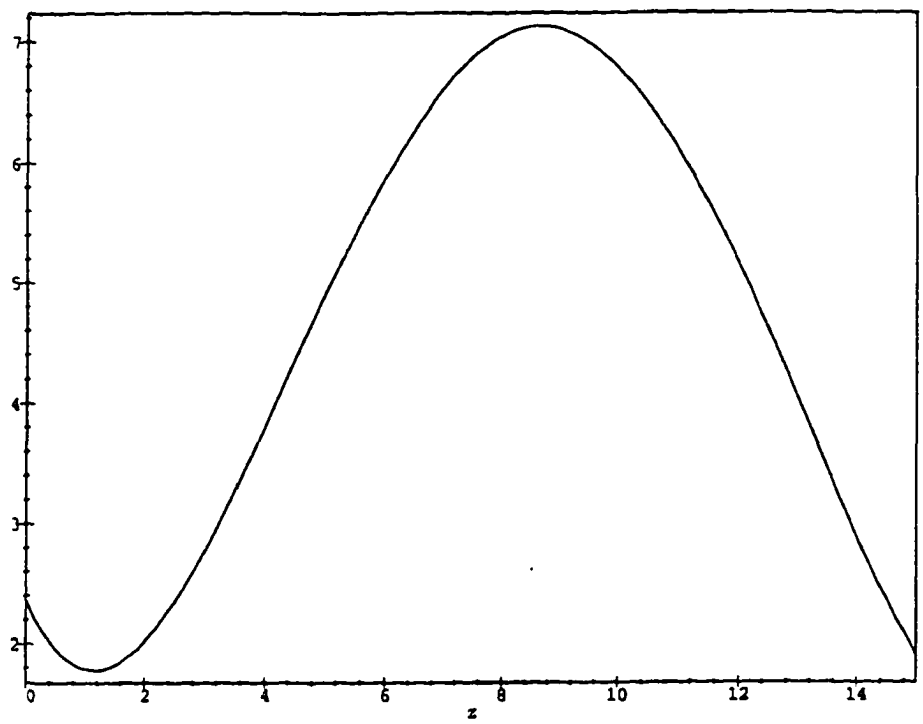

Figure 6.4: Thermal Flux from Polynomial Interpolation

6.b.2.d. Location by Functional Interpolation

However, a more accurate approximation of the axial flux is obtained by a functional interpolation of the RRAP data. This data is the average flux in a region and it can be mathematically expressed as:

$$\langle \phi(z) \rangle_i = \frac{1}{b-a} \int_{z=a}^{z=b} dz \phi(z) \quad \text{i=I..V} \quad \text{(Eq. 6.6)}$$

Where the quantity in brackets is the average flux from MCNP, b is the upper axial bound on the region and a is the lower axial bound on the region. For all the functionals in this work, the quantity 'b-a' is a constant (three inches).

A trial function needs to be chosen to represent the axial flux. This trial function is then inserted into the above set of equations so it is necessary that this trial function have five unknown coefficients. A general form for this trial function is:

$$\phi(z) = a_0 f_0(z) + a_1 f_1(z) + a_2 f_2(z) + a_3 f_3(z) + a_4 f_4(z) \quad \text{(Eq. 6.7)}$$

As was mentioned previously, the basis functions that comprise this trial function must be linearly independent. In addition, the functionals must be linearly independent in the algebraic conjugate space. For the five functionals used in this problem, it is clear that they are linearly independent in the algebraic conjugate space by referring to the definition of linear independence of functionals. The five functionals are all integrals operating on a portion of the region of interest. Since none of these regions overlap, the functionals are linearly independent.

A number of combinations of basis functions were investigated. These basis functions are tabulated below. For the hyperbolic and the trigonometric functions, a constant 'k' was introduced into the arguments of the transcendental functions. The value of this constant was chosen so that the flux would peak around the axial midplane of the core. Various other values of this constant introduce spatial variations that are not

physically reasonable. For example, choosing the value of unity causes oscillations of the flux, including negative values for the flux.

The basis functions were chosen with an approximate shape of the axial flux in mind. It was assumed that the flux would be of the form of a shifted cosine with the peak occurring slightly below the axial midplane. Simple polynomials and trigonometric functions were obvious choices for the basis functions. A flux representation including hyperbolic functions was investigated to determine the effect of these basis functions on the axial flux representation.

Trial	$f_0(z)$	$f_1(z)$	$f_2(z)$	$f_3(z)$	$f_4(z)$
i	1	sin(kz)	cos(kz)	zsin(kz)	zcos(kz)
ii	1	sin(kz)	cos(kz)	sin(2kz)	cos(2kz)
iii	1	z	z^2	sin(kz)	cos(kz)
iv	1	sin(kz)	cos(kz)	sinh(kz)	cosh(kz)
v	1	z	z^2	z^3	z^4
vi	1	z	$\frac{1}{2}(3z^2-1)$	$\frac{1}{2}(5z^3-3z)$	$\frac{1}{8}(35z^4-30z^2+3)$
vii	1	z	$2z^2-1$	$4z^3-3z$	$8z^4-8z^2+1$

Table 6.5: Trial Functions for Functional Interpolation

A special note should be made about trials vi (Legendre polynomials) and vii (Chebyshev polynomials). Since both of these sets of polynomials are defined on the range of

$-1 \le z \le 1$, the coordinate system had to be adjusted along with the definitions of the functionals. These adjustments were made to ensure consistent results. In order to ensure that these adjustments were made correctly, the strictly polynomial basis functions (trial v) were analyzed for both sets of coordinates. The results (flux shape, location of the sample from the 1969 results, the location of the maximum flux, and the value of the maximum flux) were all in agreement.

The chosen flux representation was inserted into the set of functionals. Explicitly, these functionals are defined as:

$$L_1(\ ) = \frac{1}{3}\int_0^3 dz \cdot \quad \text{(Eq. 6.8)}$$

$$L_2(\ ) = \frac{1}{3}\int_3^6 dz \cdot \quad \text{(Eq. 6.9)}$$

$$L_3(\ ) = \frac{1}{3}\int_6^9 dz \cdot \quad \text{(Eq. 6.10)}$$

$$L_4(\ ) = \frac{1}{3}\int_9^{12} dz \cdot \quad \text{(Eq. 6.11)}$$

$$L_5(\ ) = \frac{1}{3}\int_{12}^{15} dz \cdot \quad \text{(Eq. 6.12)}$$

Once the chosen flux representation is operated on by these functionals, a set of five constraint equations with five unknowns is again produced. By performing an analysis identical to the analysis done for polynomial interpolation, the location of the 1969

sample, the location of the maximum flux and the value of the maximum flux were all computed and are presented in the following tables (first, the total flux and then the thermal flux). As a reminder, the transcendental basis functions contain a constant. The value for this constant was taken to be $2\pi/15$. This number was chosen since the expected form of the axial flux is approximately cosine in shape with a peak near the origin. This value of the constant maximizes the cosine terms at the origin.

Trial	Sample location	Maximum location	Maximum total flux
i	3.23 in	8.56 in	10.7909×10^{13} n/cm^2 s
ii	3.30 in	8.55 in	10.7612×10^{13} n/cm^2 s
iii	3.21 in	8.57 in	10.8074×10^{13} n/cm^2 s
iv	3.20 in	8.57 in	10.8074×10^{13} m/cm^2 s
v	3.30 in	8.54 in	10.7560×10^{13} n/cm^2 s
vi	3.30 in	8.54 in	10.7560×10^{13} n/cm^2 s
vii	3.30 in	8.54 in	10.7560×10^{13} n/cm^2 s

Table 6.6: Total Flux Results from Functional Interpolation

Trial	Sample location	Maximum location	Maximum thermal flux
i	3.24 in	8.69 in	7.3196×10^{13} n/cm^2 s
ii	3.33 in	8.71 in	7.2934×10^{13} n/cm^2 s
iii	3.23 in	8.68 in	7.3214×10^{13} n/cm^2 s
iv	3.21 in	8.68 in	7.3224×10^{13} m/cm^2 s
v	3.32 in	8.66 in	7.2838×10^{13} n/cm^2 s
vi	3.32 in	8.66 in	7.2838×10^{13} n/cm^2 s
vii	3.32 in	8.66 in	7.2838×10^{13} n/cm^2 s

Table 6.7: Thermal Flux Results from Functional Interpolation

There is remarkably good agreement in all of the values for the total flux. There is about a four percent variation in the calculations for the 1969 sample location. In addition, the plots of these trial functions for the total axial flux are remarkably similar as seen in Figures 6.5 through 6.11. The only major difference in the plots is that a few of these plots have an obvious inflection point near the top of the core. This upturn is evident in trial ii and all the purely polynomial trials. The results for the thermal flux are similar to those of the total flux. Figure 6.12 is an example of the thermal flux found from functional interpolation using a simple polynomial (trial v).

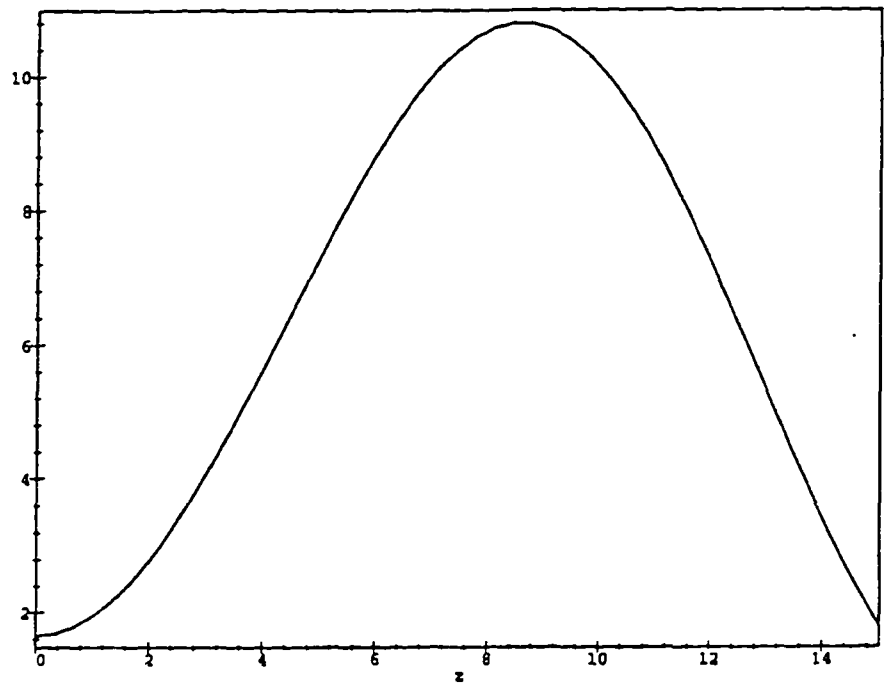

Figure 6.5: Total Flux from Functional Interpolation, Trial i

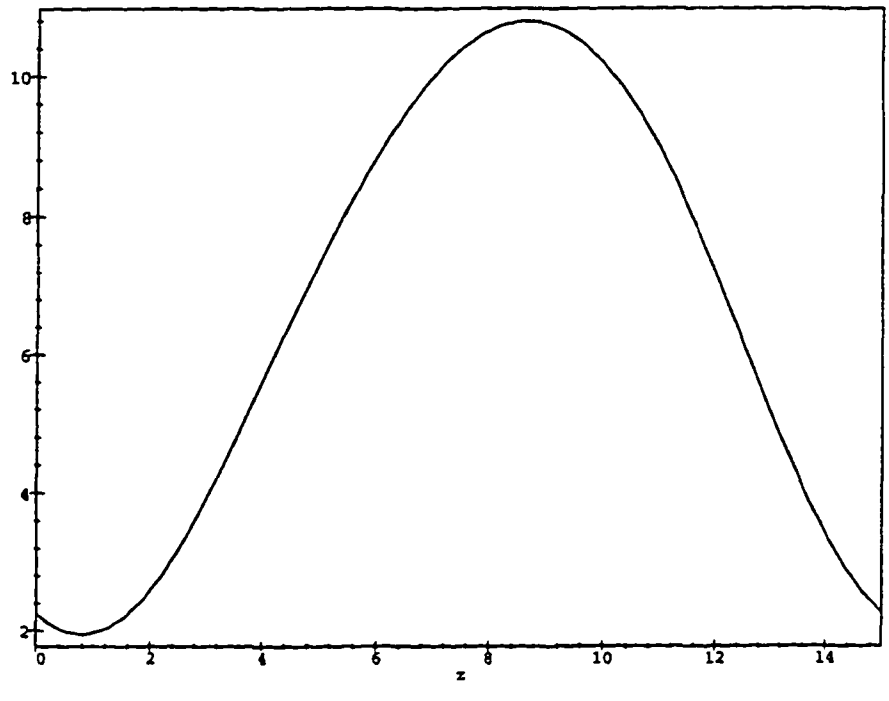

Figure 6.6: Total Flux from Functional Interpolation, Trial ii

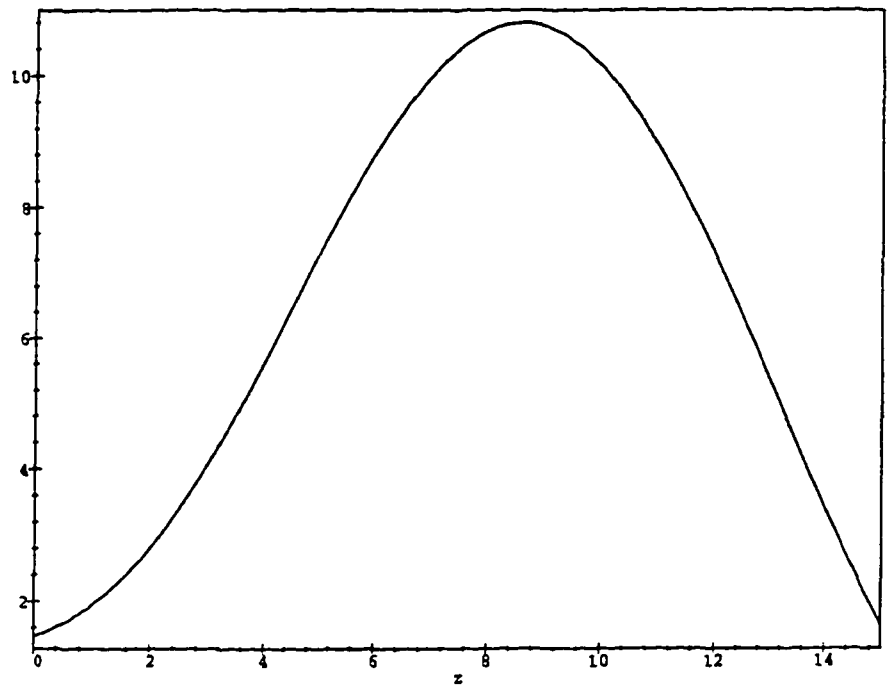

Figure 6.7: Total Flux from Functional Interpolation, Trial iii

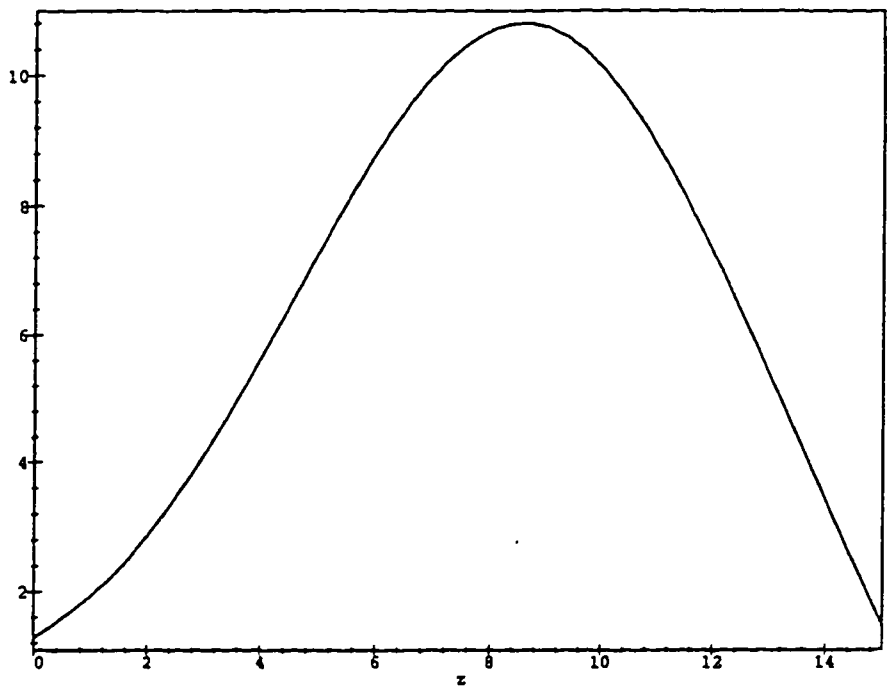

Figure 6.8: Total Flux from Functional Interpolation, Trial iv

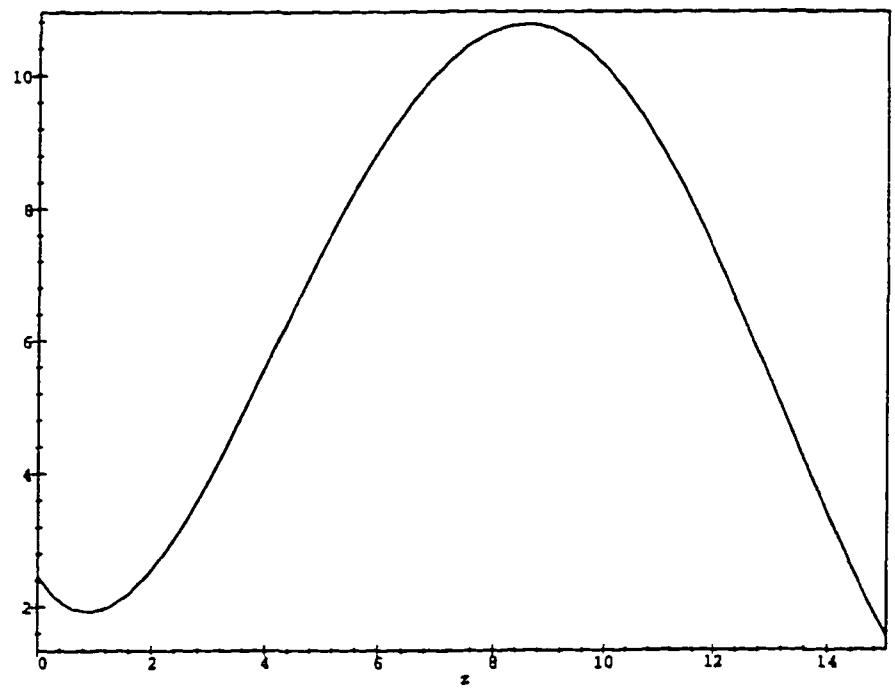

Figure 6.9: Total Flux from Functional Interpolation, Trial v

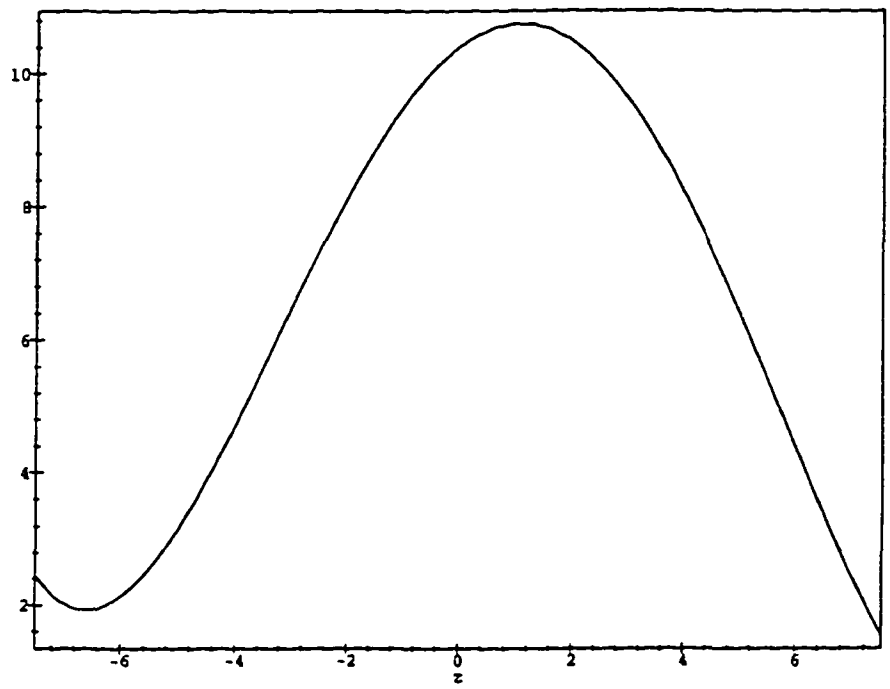

Figure 6.10: Total Flux from Functional Interpolation, Trial vi

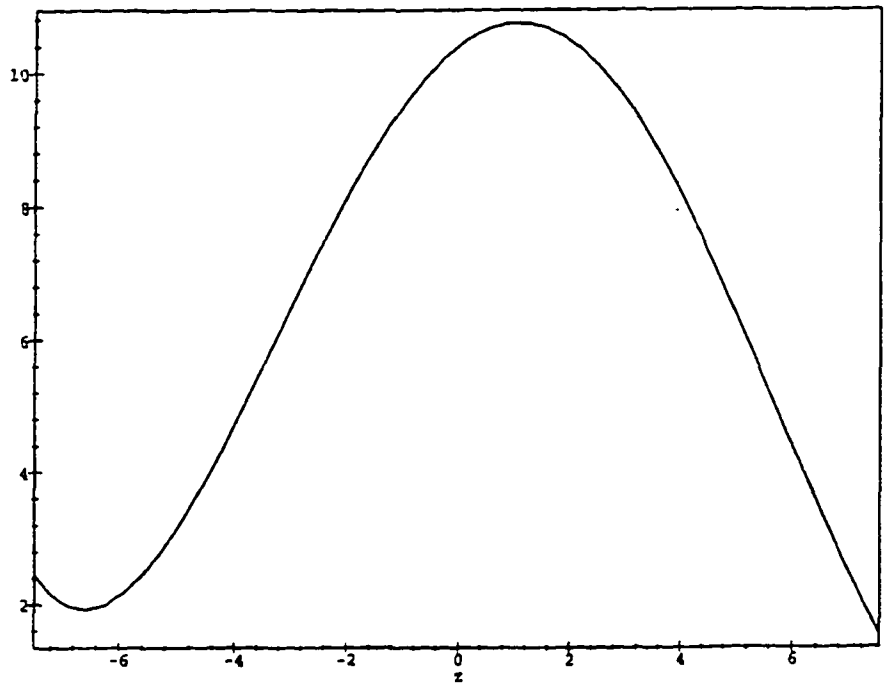

Figure 6.11: Total Flux from Functional Interpolation, Trial vii

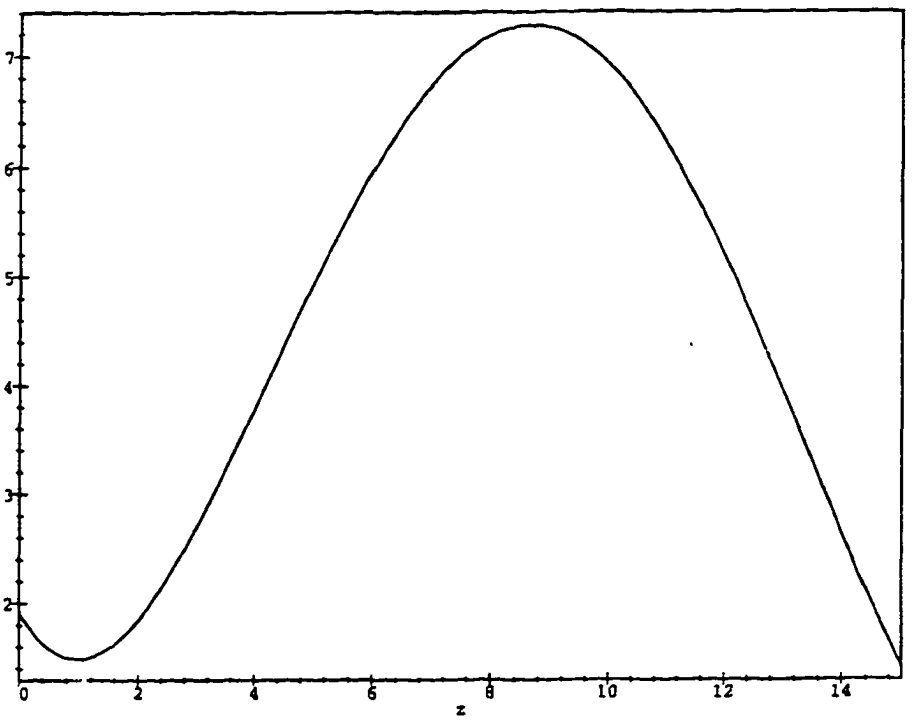

Figure 6.12: Thermal Flux from Functional Interpolation, Trial v

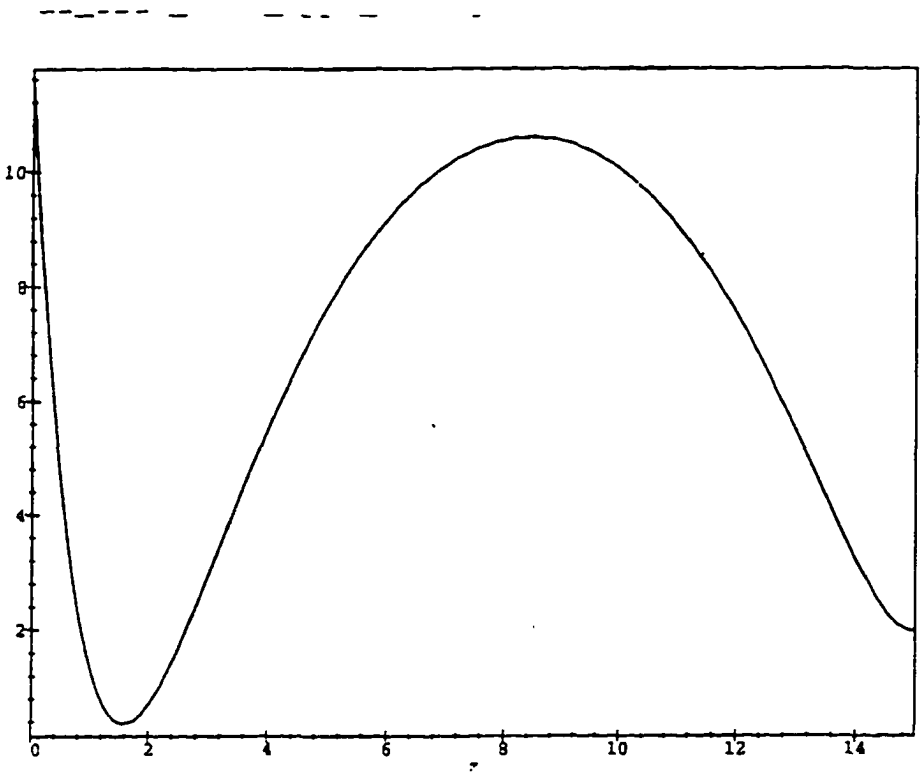

Figure 6.13: Thermal Flux from Functional Interpolation, Trial iv

Considering only those trial functions that are strictly polynomials (v. vi. and vii). the results are identical for all three cases for both the thermal and the total fluxes. The polynomial that interpolates a set of point data (pointwise polynomial interpolation) is known to be unique.[1] As was presented in the discussion of the theory of functional interpolation, there exists a similar theorem for the uniqueness of a solution from functional polynomial interpolation. The results from trial v. vi an vii numerically confirm this theorem. This can be proven analytically by showing that the generalized Gram determinant is non-zero.

The results from these reconstruction trials imply that there is only a weak dependence on the choice of basis functions provided that engineering knowledge is applied to the choice of the basis functions. For example, the choice of purely hyperbolic basis functions produces an unphysical representation of the flux. A plot of the reconstructed flux using hyperbolic basis functions reveals a shape similar to those found with other basis functions but there is a much larger up turn in the flux near the top of the core (see Figure 6.13).

Any attempt to determine a more 'accurate' location of the sample is specious since the results from the functional interpolation trials all indicate that the sample is located within a one tenth of an inch region. Because the sample will occupy some finite volume, any further refinement of the location would require a more detailed calculation involving the size of the sample. The purpose of this investigation was to determine if the sample was located at the position of the peak flux. The results indicate that it was not located at the center of the core.

A comparison of the results from the linear interpolation (~2.95 in), the polynomial interpolation (~3.25 in), and the functional interpolation (~3.25 in) indicate that the only major difference between the three methods is the location found from linear interpolation. Even this result is within ten percent of the other two methods. The results are curious since it was expected that the functional interpolation results would be more 'accurate' than the polynomial interpolation results since there was an assumption made concerning the location of the interpolation points for the polynomial method. It could be that these results are somewhat similar due to the limited number of data points in this problem.

This application of interpolation theory allowed the calculation of the location of a sample in the TRIGA reactor but the methodology has much farther reaching implications. One of the main drawbacks to any Monte Carlo code is the length of time needed to run the code. Part of this run time is due to a detailed geometry. A simpler geometry implies a quicker Monte Carlo code run. Currently, if a detailed flux distribution is desired, then the geometry must be very much sub-divided. This work would allow a less detailed geometry to be used. This would imply a quicker execution of the code. The detailed flux distribution would then be created by a functional interpolation of the Monte Carlo results not from a finely divided geometry.

Another advantage to applying functional interpolation to the Monte Carlo results would be the creation of an analytic form for the flux. The results from Monte Carlo are similar to a histogram: The flux in each region is represented by a number for that particular volume. With purely numerical results from Monte Carlo, any subsequent

calculations involving the flux must necessarily be performed numerically. An analytic form for the flux would allow the analytic calculation of currents, reaction rates, and doses to name a few of the possibilities. This could greatly reduce the computational time needed to perform many of the tasks in nuclear engineering design.

Ideally, the results from this investigation should have been confirmed experimentally by physically moving the central thimble further down into the core. Unfortunately, immediately after this work had been performed, the configuration and the material composition of the TRIGA core were radically altered.[20] This alteration prevented the experimental verification of these results.

7. Conclusions and Recommendations

The main purpose of this work was to establish a methodology to analytically determine the components of the errors associated with nodal flux reconstruction. The chosen method was very simple in concept and it provided a number with which to compare various reconstruction methods. The error due to the functional interpolation method was 'bounded' by examining a series of probable two-dimensional flux shapes. For simple polynomial flux shapes, the functional interpolation error was determined to be zero. This implied that any numeric error must have been caused by the nodal input data for these simple flux shapes.

The effects of the errors in the nodal input data were analytically determined and presented as two-dimensional shapes. These shapes indicated how the errors in the nodal input data propagated throughout the node. For some of the input data, the propagated error actually increased inside the node. Obviously, an accurate nodal reconstruction method would require very precise nodal input data.

An indication of the "goodness" of a nodal reconstruction scheme was developed. This Figure of Merit incorporates the effects of the errors due to the functional interpolation method and the effects of the errors of the nodal input data. Since the devised Figure of Merit is a combined result of the two errors, a decrease in one of these errors will decrease the Figure of Merit. Therefore, it is necessary to search for an improved reconstruction scheme that minimizes the Figure of Merit.

Unfortunately, it is in this section that the limitations of proposed error analysis are found. The method for determining the Figure of Merit was found to be limited to those reconstruction schemes that have only one energy group. This is indeed disappointing since there are a number of methods which incorporate all of the energy groups in reconstructing the two-dimensional flux.

In addition, the determination of the Figure of Merit was found to have a few calculations limitations. It was initially desired to have a methodology that would be robust in the sense that the method would give an absolute evaluation of the accuracy of the reconstruction scheme. However, at present, the symbolic manipulation software available is unable to invert a symbolically dense twenty-one by twenty-one matrix with general hyperbolic (or trigonometric) entries. This forced some simplifications such as restricting the analysis to square nodes and to utilizing the material properties from a benchmark problem.

Finally, and perhaps most importantly, it was desired to apply the nodal flux reconstruction methodology to other problems. A few examples of the application of these methods have been successfully presented in this work. These examples were directly related to the study of the neutron population. Using this method, it is possible to determine if a particular region should be analyzed by a diffusion or transport approach. It has also been shown that the methodology of nodal flux reconstruction can provide additional, functional information from Monte Carlo results.

However, it is also desired to show that this method is not simply a method for reactor analysis. The idea of functional interpolation is not just restricted to nodal flux

reconstruction. In part, it is the application of functional interpolation that leads directly to recommendations for further work:

1. It was shown that the nodal input error propagates throughout the node. The method to determine this error propagation can be applied in a similar manner to finite difference and finite element calculations since both of these methods are dependent upon "input" data. By input data, it is meant the specified edge values. If it is assumed that these edge values have an associated error, then the methodology developed in this work could be applied to determine how this edge error propagates through (or into) adjacent points or elements. The errors associated with the boundary conditions (edge values) should not uniformly apply through the region of interest. It is expected that an analogous situation with the nodal input error propagation should occur.

2. The chosen Figure of Merit may not be the optimal choice. As it is defined, the Figure of Merit has an obvious weakness that is demonstrated by the analysis of Koebke and Wagner's reconstruction method. The Figure of Merit is dominated by the functional interpolation error from two terms. A large Figure of Merit could indicate a poor reconstruction scheme when in actuality the bulk of the error is caused by one or two cases. Additional research needs to be done to determine the most equitable method for determining the Figure of Merit.

3. It might be possible to formally prove that pointwise, multi-dimensional interpolation is a simplification of functional interpolation with a particular choice of functionals. Additional, it would be of great interest if a form of interpolation could be developed that was more general than functional interpolation. Functional interpolation

allows for the interpolation of pieces of data that are functionally constant throughout the region of interest. A more general form of interpolation would be able to interpolate functional information; that is, information that is not functionally constant.

4. Functional interpolation is applicable whenever there exists a set of data that is functionally constant over a region of interest. This implies a huge variety of fields such as tomography, cross section generation, statistical analysis, dosimetry, and pollution migration to name a few areas. The application of functional interpolation to these areas should allow the creation of more detailed distributions than are currently available through current data analysis techniques. For example, functional interpolation could be used to create two- and three-dimensional distributions (pictures) from one- and two-dimensional distributions. A photograph could be decomposed into pixels. Each pixel occupies a finite area and this area is assigned a specific color (wavelength or intensity). Functionally interpolating a region of pixels would allow the creation of two- or three-dimensional analytic distributions.

5. Finally, and perhaps most importantly, this work has implications concerning the future of numerical analysis. Currently, numerical analysis is utilized on a variety of problems because an analytical solution is not readily available. With the advent of symbolic manipulators coupled with more powerful computers, many of the problems tackled by numerical analysis may be 'analytically' determined by a symbolic manipulator. This work has shown that a symbolic manipulator can analyze, in depth, a numerical analysis method. It stands to reason that improvements in hardware and

software will lead to the elimination of using numerical analysis for many types of problems.

Appendix I: Explicit Representations and Graphs of the Errors due to the Nodal Input Data

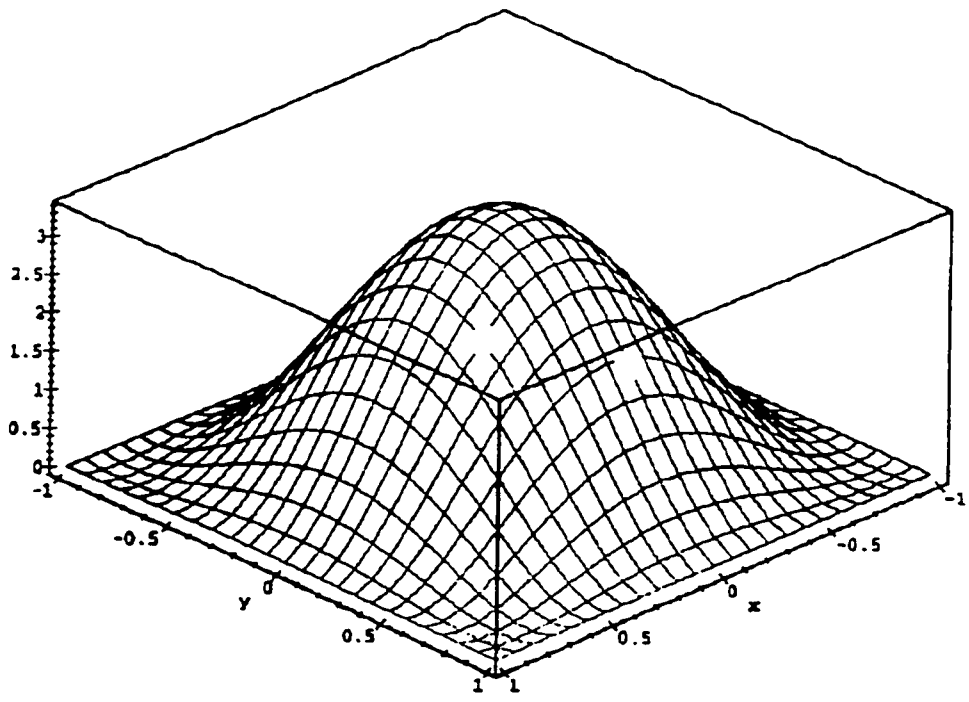

Figure I.1: Error shape due to node average flux

$$\frac{1}{4ab}\int_{-a}^{a} dx \int_{-b}^{b} dy\phi(x,y) \Rightarrow r_1 = \frac{1}{64}\begin{pmatrix} 216 - 396\xi^2 - 396\eta^2 + 180\xi^4 + 180\eta^4 \\ -180\xi^4\eta^2 - 180\xi^2\eta^4 + 576\xi^2\eta^4 \end{pmatrix}$$

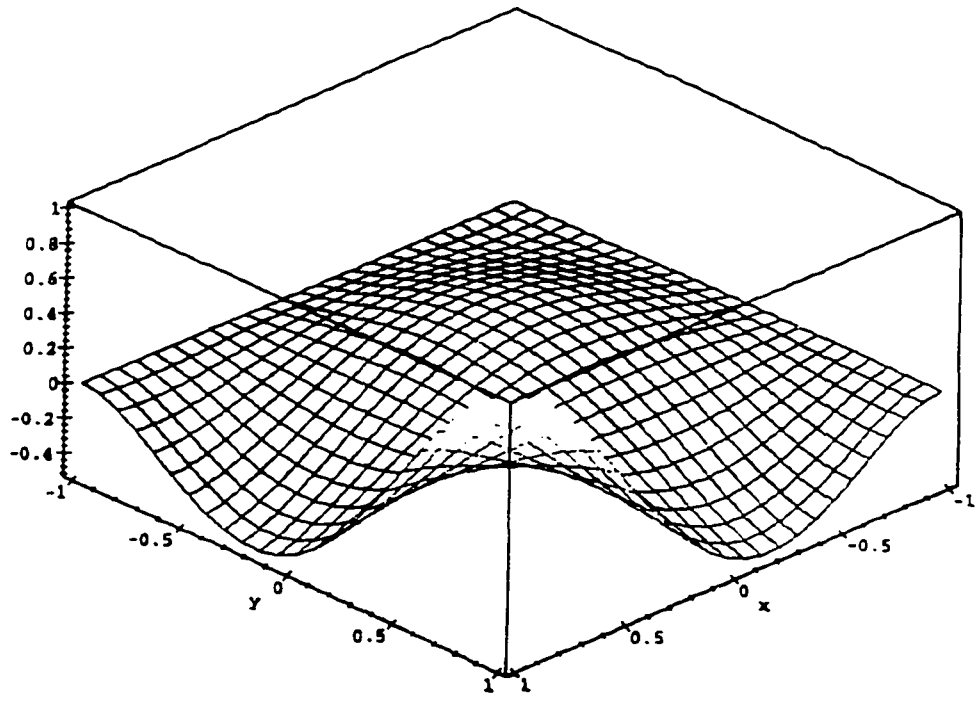

Figure I.2: Error shape due to the corner flux at (a,b)

$$\phi(a,b) \Rightarrow r_2 = \frac{1}{64}\begin{pmatrix} -30\eta\xi^4 + 4\eta^3 - 39\xi^2 + 32\eta\xi + 4\xi^3 - 8\eta^3\xi - 30\eta^4\xi + 72\eta\xi^2 \\ +15\eta^4 + 72\eta^2\xi + 15\xi^4 - 39\eta^2 - 18\xi - 12\eta^3\xi^2 - 18\eta - 45\eta^4\xi^2 \\ +144\eta^2\xi^2 - 8\eta\xi^3 + 10 - 45\eta^2\xi^4 - 12\eta^2\xi^3 \end{pmatrix}$$

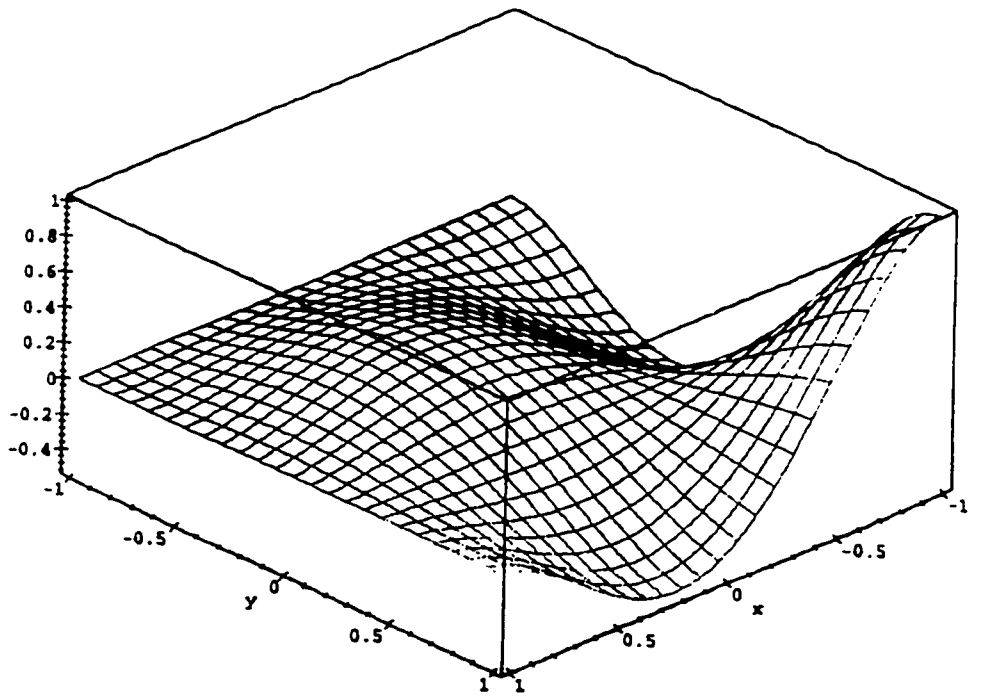

Figure I.3: Error shape due to the corner flux at (-a,b)

$$\phi(-a,b) \Rightarrow r_3 = \frac{1}{64}\begin{pmatrix} 30\eta\xi^4 - 4\eta^3 - 39\xi^2 - 32\eta\xi + 4\xi^3 + 8\eta^3\xi - 30\eta^4\xi - 72\eta\xi^2 \\ +15\eta^4 + 72\eta^2\xi + 15\xi^4 - 39\eta^2 - 18\xi + 12\eta^3\xi^2 + 18\eta - 45\eta^4\xi^2 \\ +144\eta^2\xi^2 + 8\eta\xi^3 + 10 - 45\eta^2\xi^4 - 12\eta^2\xi^3 \end{pmatrix}$$

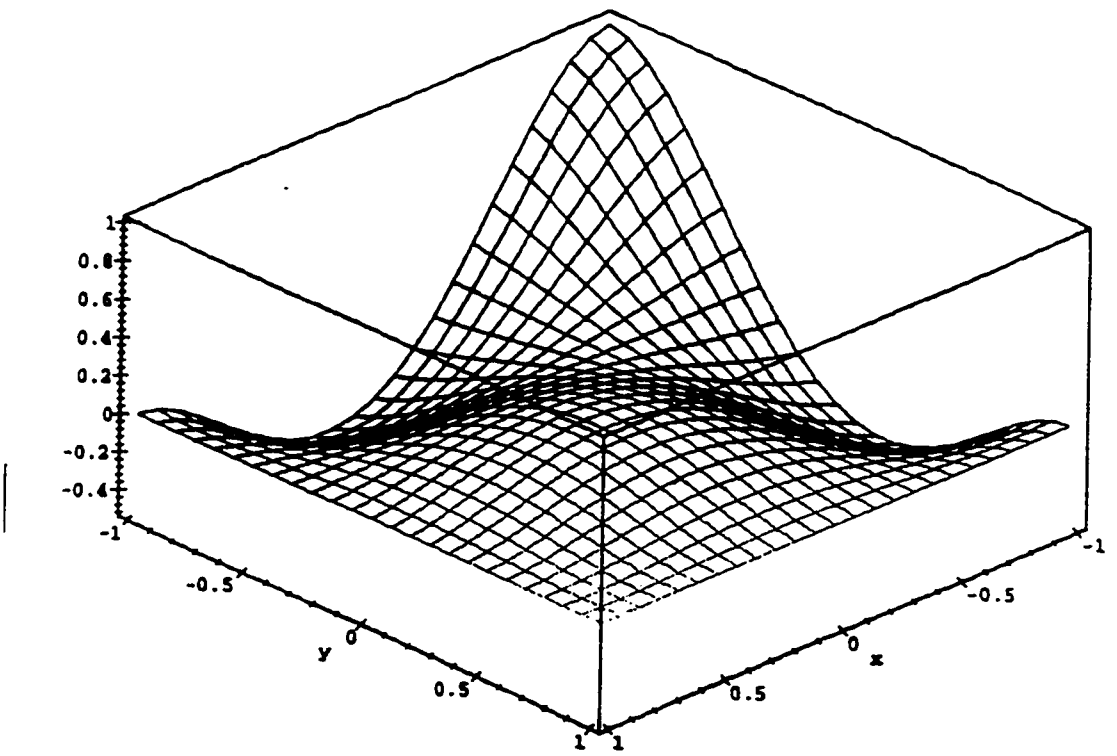

Figure I.4: Error shape due to the corner flux at (-a,-b)

$$\phi(-a,-b) \Rightarrow r_4 = \frac{1}{64}\begin{pmatrix} 30\eta\xi^4 - 4\eta^3 - 39\xi^2 + 32\eta\xi - 4\xi^3 - 8\eta^3\xi + 30\eta^4\xi - 72\eta\xi^2 \\ +15\eta^4 - 72\eta^2\xi + 15\xi^4 - 39\eta^2 + 18\xi + 12\eta^3\xi^2 + 18\eta - 45\eta^4\xi^2 \\ +144\eta^2\xi^2 - 8\eta\xi^3 + 10 - 45\eta^2\xi^4 + 12\eta^2\xi^3 \end{pmatrix}$$

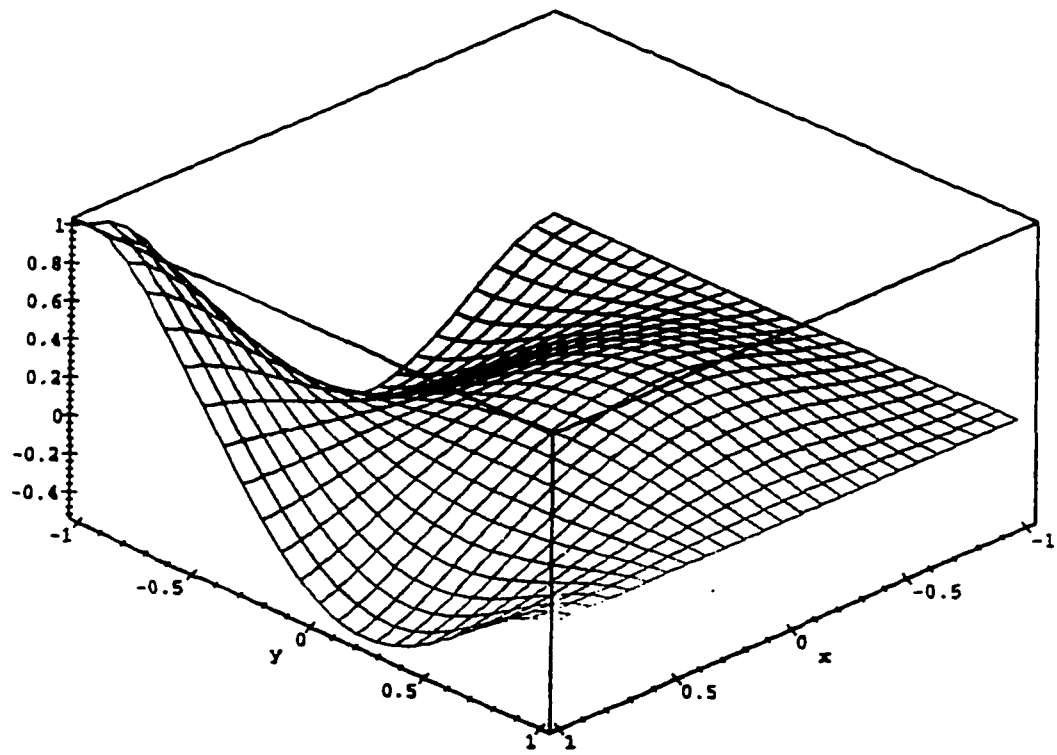

Figure I.5: Error shape due to the corner flux at (a,-b)

$$\phi(a,-b) \Rightarrow r_5 = \frac{1}{64}\begin{pmatrix} 30\eta\xi^4 - 4\eta^3 - 39\xi^2 + 32\eta\xi - 4\xi^3 - 8\eta^3\xi + 30\eta^4\xi - 72\eta\xi^2 \\ +15\eta^4 - 72\eta^2\xi + 15\xi^4 - 39\eta^2 + 18\xi + 12\eta^3\xi^2 + 18\eta - 45\eta^4\xi^2 \\ +144\eta^2\xi^2 - 8\eta\xi^3 + 10 - 45\eta^2\xi^4 + 12\eta^2\xi^3 \end{pmatrix}$$

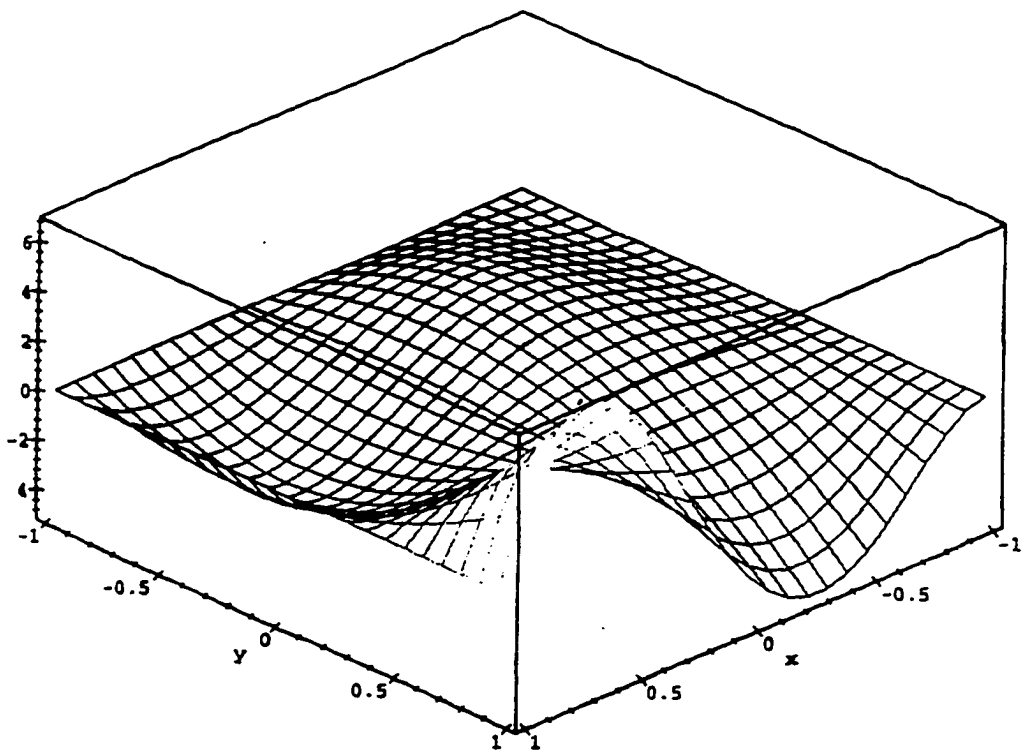

Figure I.6: Error shape due to the x-directed corner current at (a,b)

$$-D_g \frac{\partial \phi(x,y)}{\partial x}\bigg|_{\substack{x=a \\ y=b}} \Rightarrow r_6 = \frac{a}{64D_g}\begin{pmatrix} -4\eta+1-10\eta^4\xi+8\eta\xi-15\eta^4\xi^2-8\eta^3\xi-6\eta^3-3\xi^2 \\ +4\eta^3+5\eta^4+12\eta^2\xi+12\eta\xi^2+18\eta^2\xi^2-2\xi-12\eta^3\xi^2 \end{pmatrix}$$

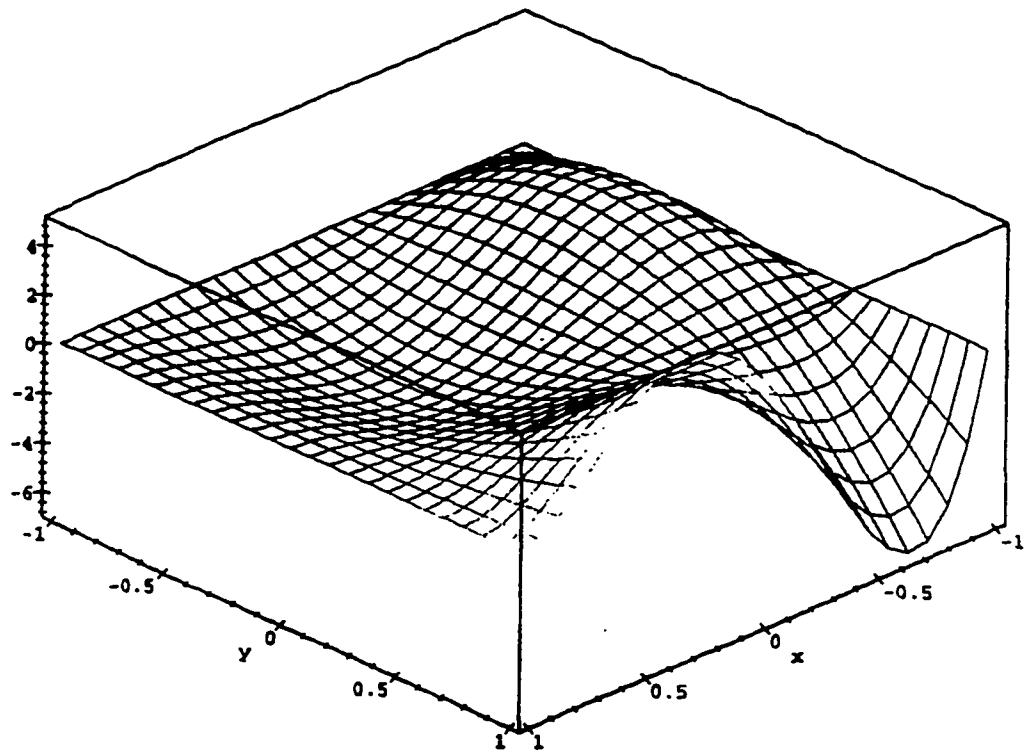

Figure I.7: Error shape due to the x-directed corner current at (-a,b)

$$-D_g \left.\frac{\partial\, \phi(x,y)}{\partial\, x}\right|_{\substack{x=-a \\ y=b}} \Rightarrow r_7 = \frac{a}{64 D_g}\begin{pmatrix} -12\eta^2\xi + 10\eta^4\xi + 4\eta^3 + 15\eta^4\xi^2 - 5\eta^4 - 4\eta + 8\eta\xi - 8\eta^3\xi \\ -12\eta^3\xi^2 + 2\xi + 3\xi^2 + 6\eta^2 + 12\eta\xi^2 - 1 - 18\eta^2\xi^2 \end{pmatrix}$$

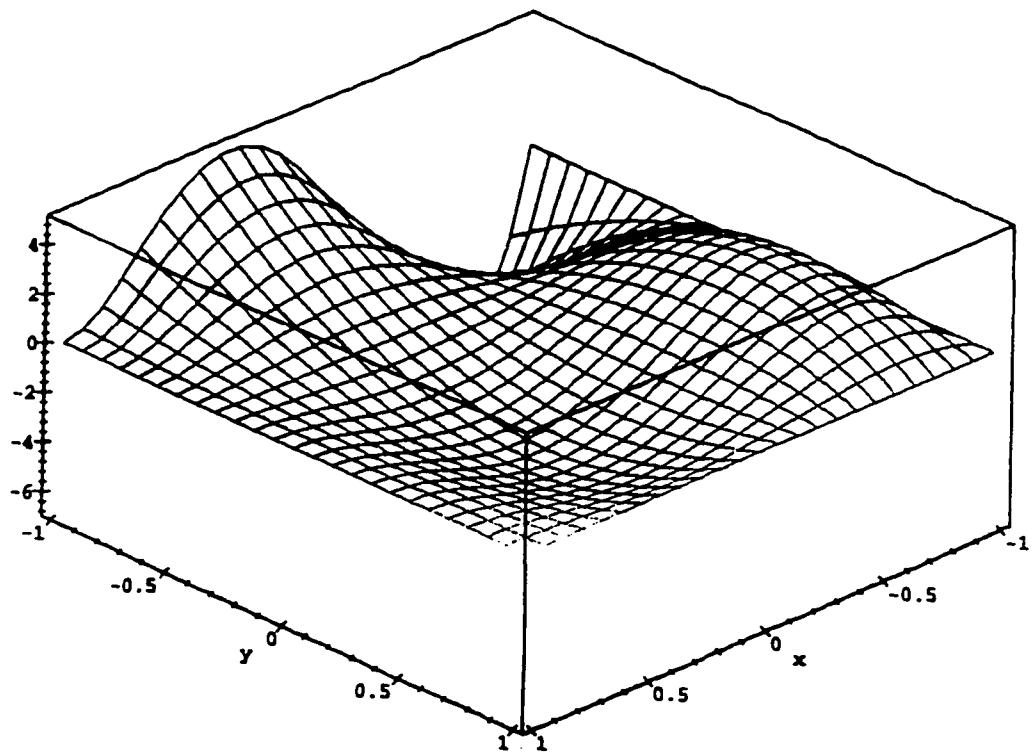

Figure I.8: Error shape due to the x-directed corner current at (-a,-b)

$$-D_g \frac{\partial\, \phi(x,y)}{\partial\, x}\Big|_{\substack{x=-a \\ y=-b}} \Rightarrow r_g = \frac{a}{64D_g}\begin{pmatrix} +12\eta^2\xi - 10\eta^4\xi + 4\eta^3 + 15\eta^4\xi^2 - 5\eta^4 - 4\eta - 8\eta\xi + 8\eta^3\xi \\ -12\eta^3\xi^2 - 2\xi + 3\xi^2 + 6\eta^2 + 12\eta\xi^2 - 1 - 18\eta^2\xi^2 \end{pmatrix}$$

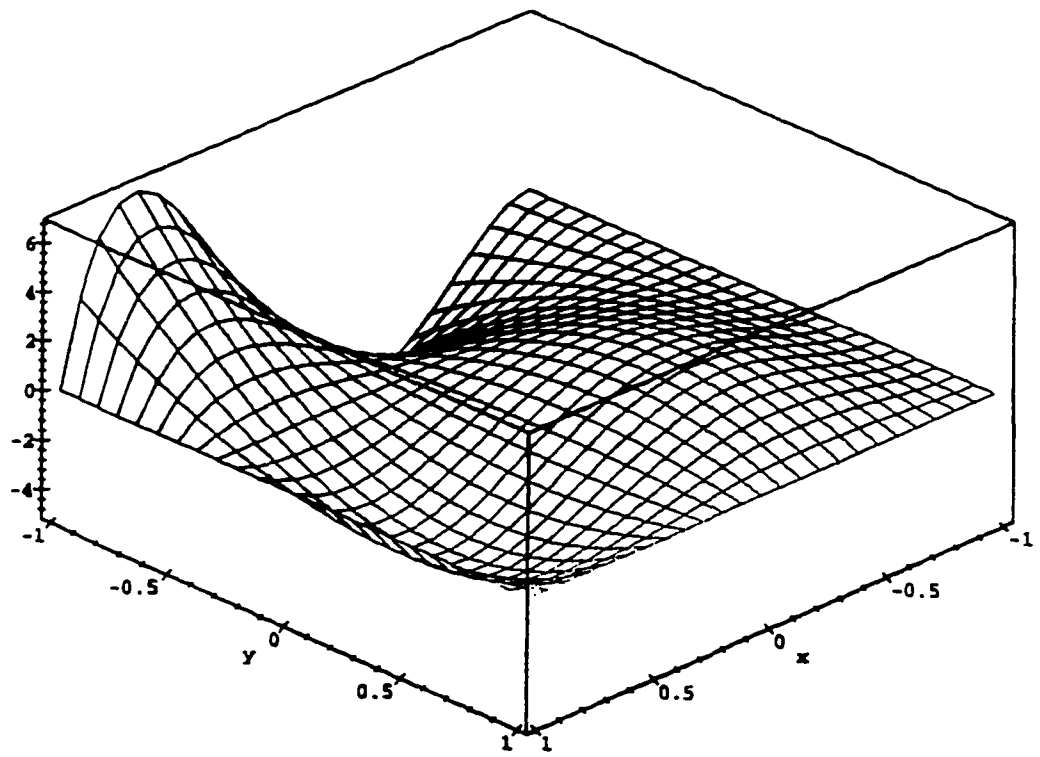

Figure I.9: Error shape due to the x-directed corner current at (a,-b)

$$-D_g \left.\frac{\partial\, \phi\,(x,y)}{\partial\, x}\right|_{\substack{x=a \\ y=-b}} \Rightarrow r_9 = \frac{a}{64D_g}\begin{pmatrix} -12\eta^2\xi + 10\eta^4\xi + 4\eta^3 - 15\eta^4\xi^2 + 5\eta^4 - 4\eta - 8\eta\xi + 8\eta^3\xi \\ -12\eta^3\xi^2 + 2\xi - 3\xi^2 - 6\eta^2 + 12\eta\xi^2 + 1 + 18\eta^2\xi^2 \end{pmatrix}$$

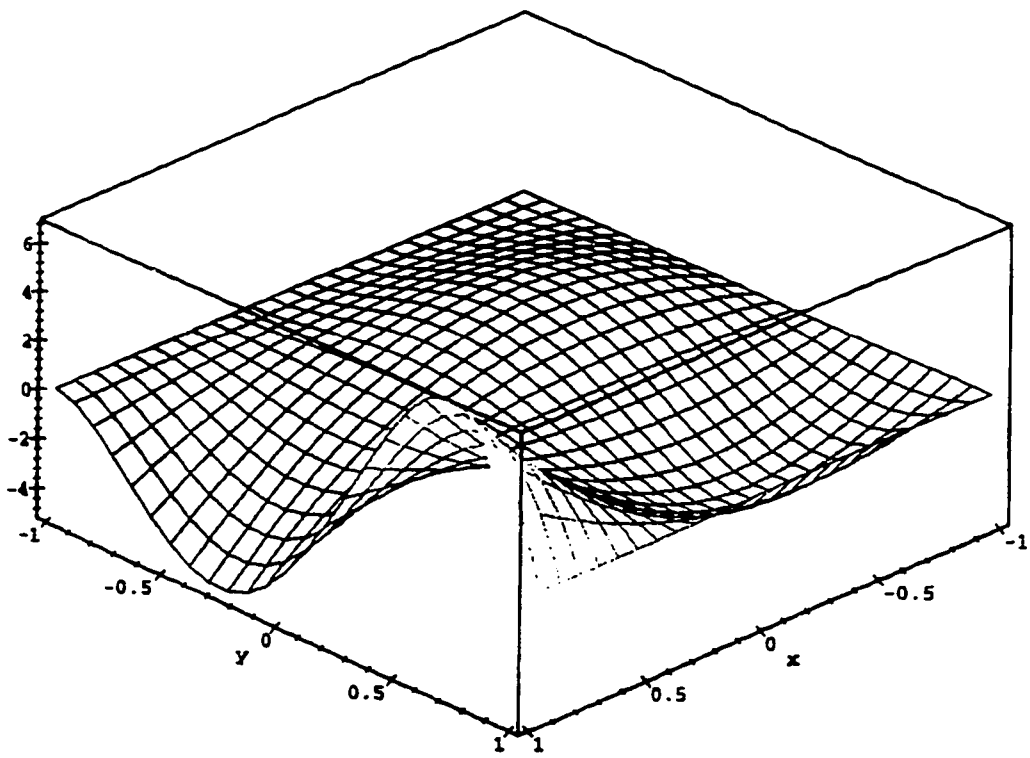

Figure I.10: Error shape due to the y-directed corner current at (a,b)

$$-D_g \left.\frac{\partial\ \phi(x,y)}{\partial\ y}\right|_{\substack{x=a\\y=b}} \Rightarrow r_{10} = \frac{b}{64D_g}\begin{pmatrix} +12\eta^2\xi - 10\eta\xi^4 - 6\xi^2 - 15\eta^2\xi^4 + 5\xi^4 - 2\eta + 8\eta\xi - 8\eta\xi^3 \\ -12\eta^2\xi^3 - 4\xi + 4\xi^3 - 3\eta^2 + 12\eta\xi^2 + 1 + 18\eta^2\xi^2 \end{pmatrix}$$

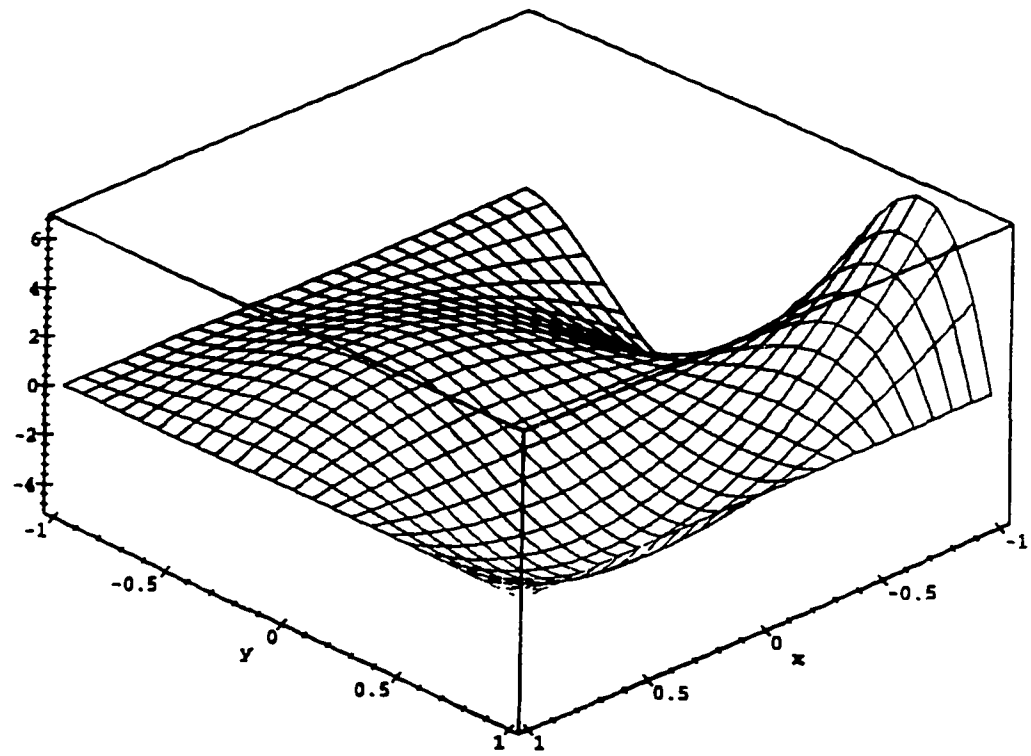

Figure I.11: Error shape due to the y-directed corner current at (-a,b)

$$-D_g \left.\frac{\partial\, \phi(x,y)}{\partial\, y}\right|_{\substack{x=-a \\ y=b}} \Rightarrow r_{11} = \frac{b}{64 D_g}\begin{pmatrix} +12\eta^2\xi + 10\eta\xi^4 - 6\xi^2 - 15\eta^2\xi^4 + 5\xi^4 + 2\eta - 8\eta\xi + 8\eta\xi^3 \\ -12\eta^2\xi^3 - 4\xi + 4\xi^3 - 3\eta^2 - 12\eta\xi^2 + 1 + 18\eta^2\xi^2 \end{pmatrix}$$

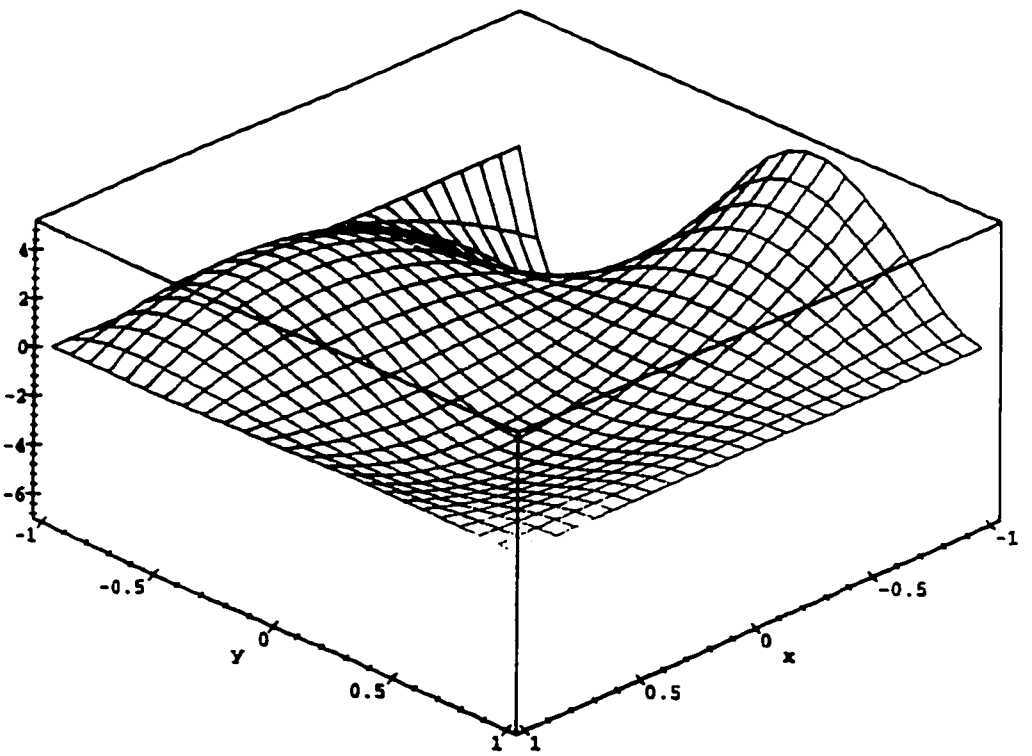

Figure I.12: Error shape due to the y-directed corner current at (-a,-b)

$$-D_g \left.\frac{\partial\, \phi\,(x,y)}{\partial\, y}\right|_{\substack{x=-a \\ y=-b}} \Rightarrow r_{12} = \frac{b}{64D_g}\begin{pmatrix} +12\eta^2\xi - 10\eta\xi^4 + 6\xi^2 + 15\eta^2\xi^4 - 5\xi^4 - 2\eta - 8\eta\xi + 8\eta\xi^3 \\ -12\eta^2\xi^3 - 4\xi + 4\xi^3 + 3\eta^2 + 12\eta\xi^2 - 1 - 18\eta^2\xi^2 \end{pmatrix}$$

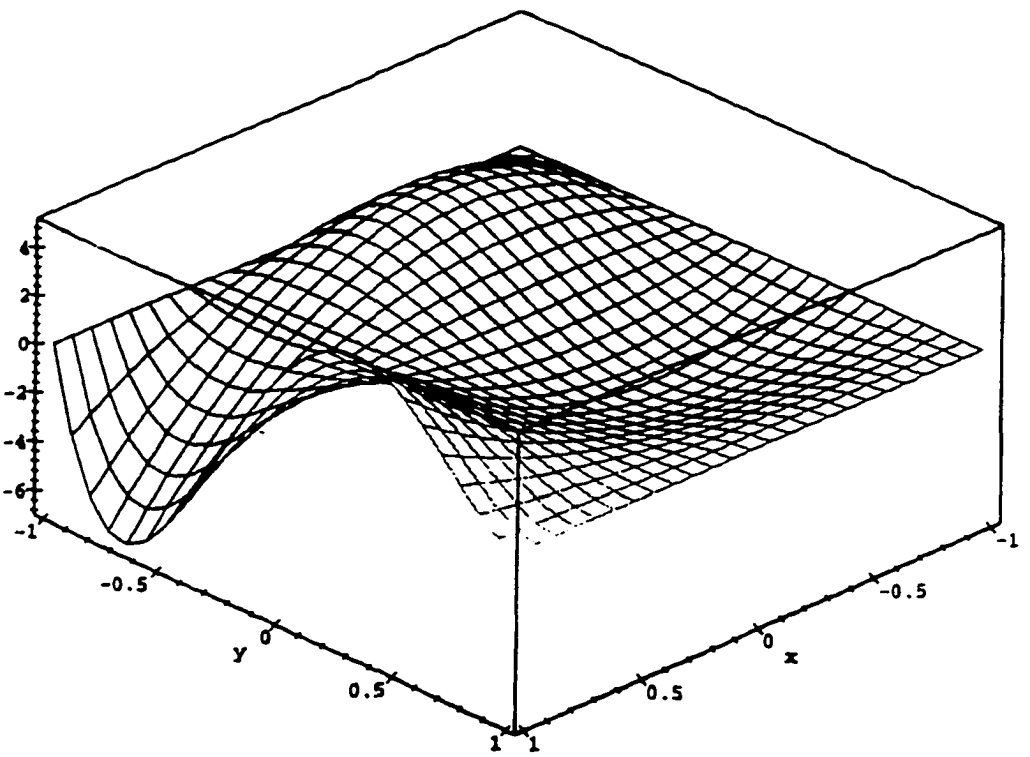

Figure I.13: Error shape due to the y-directed corner current at (a,-b)

$$-D_g \left.\frac{\partial\, \phi(x,y)}{\partial\, y}\right|_{\substack{x=a \\ y=-b}} \Rightarrow r_{13} = \frac{b}{64D_g}\begin{pmatrix} +12\eta^2\xi + 10\eta\xi^4 + 6\xi^2 + 15\eta^2\xi^4 - 5\xi^4 + 2\eta + 8\eta\xi - 8\eta\xi^3 \\ -12\eta^2\xi^3 - 4\xi + 4\xi^3 + 3\eta^2 - 12\eta\xi^2 - 1 - 18\eta^2\xi^2 \end{pmatrix}$$

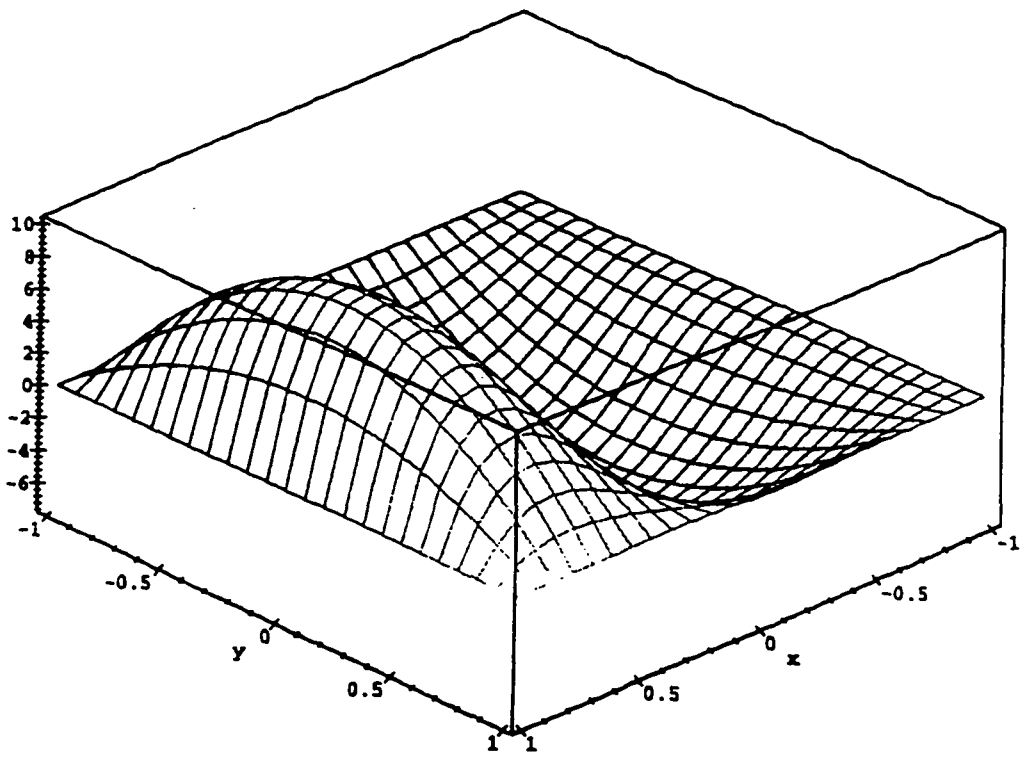

Figure I.14: Error shape due to the x-directed surface averaged current at x=a

$$\frac{-D_g}{2b}\int_{-b}^{b} dy \left.\frac{\partial \phi(x,y)}{\partial x}\right|_{x=a} \Rightarrow r_{14} = \frac{a}{64D_g}\begin{pmatrix} 24\eta - 6 + 36\eta^2 - 24\eta\xi^2 + 30\eta^4\xi^2 \\ -24\eta^3 - 36\eta^2\xi^2 + 24\eta^3\xi^2 + 6\xi^2 - 30\eta^4 \end{pmatrix}$$

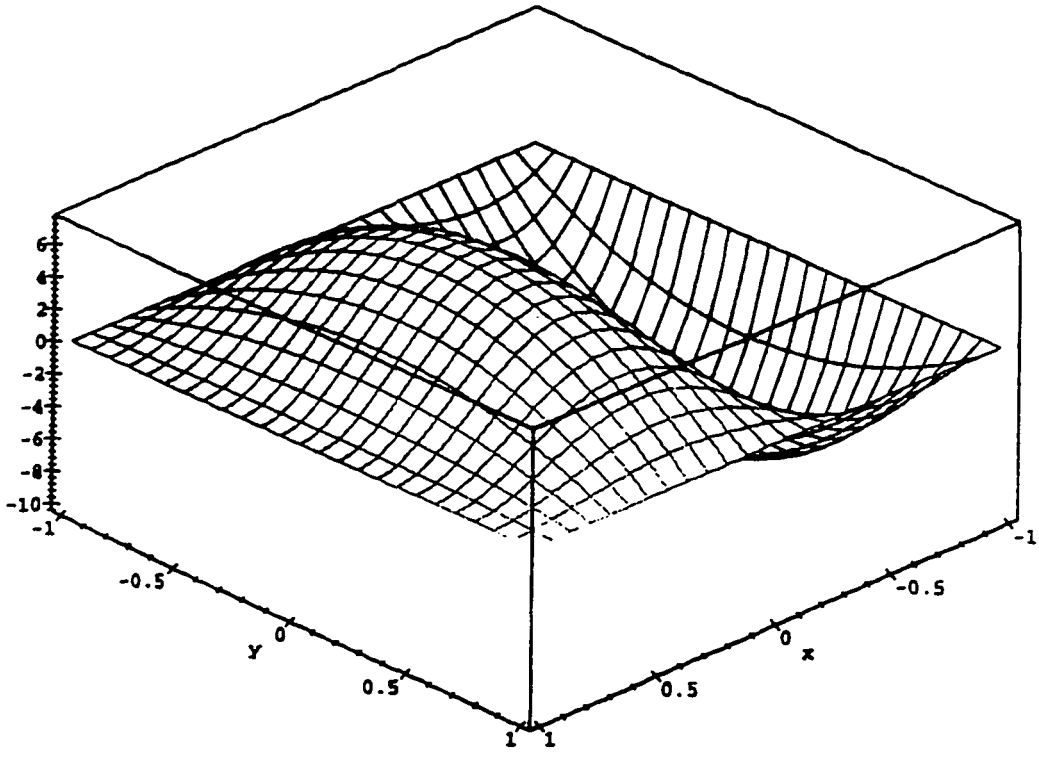

Figure I.15: Error shape due to the x-directed surface averaged current at x=-a

$$\frac{-D_g}{2b}\int_{-b}^{b} dy \left.\frac{\partial\,\phi(x,y)}{\partial\,x}\right|_{x=-a} \Rightarrow r_{15} = \frac{a}{64D_g}\begin{pmatrix} 24\eta + 6 - 36\eta^2 - 24\eta\xi^2 - 30\eta^4\xi^2 \\ -24\eta^3 + 36\eta^2\xi^2 + 24\eta^3\xi^2 - 6\xi^2 + 30\eta^4 \end{pmatrix}$$

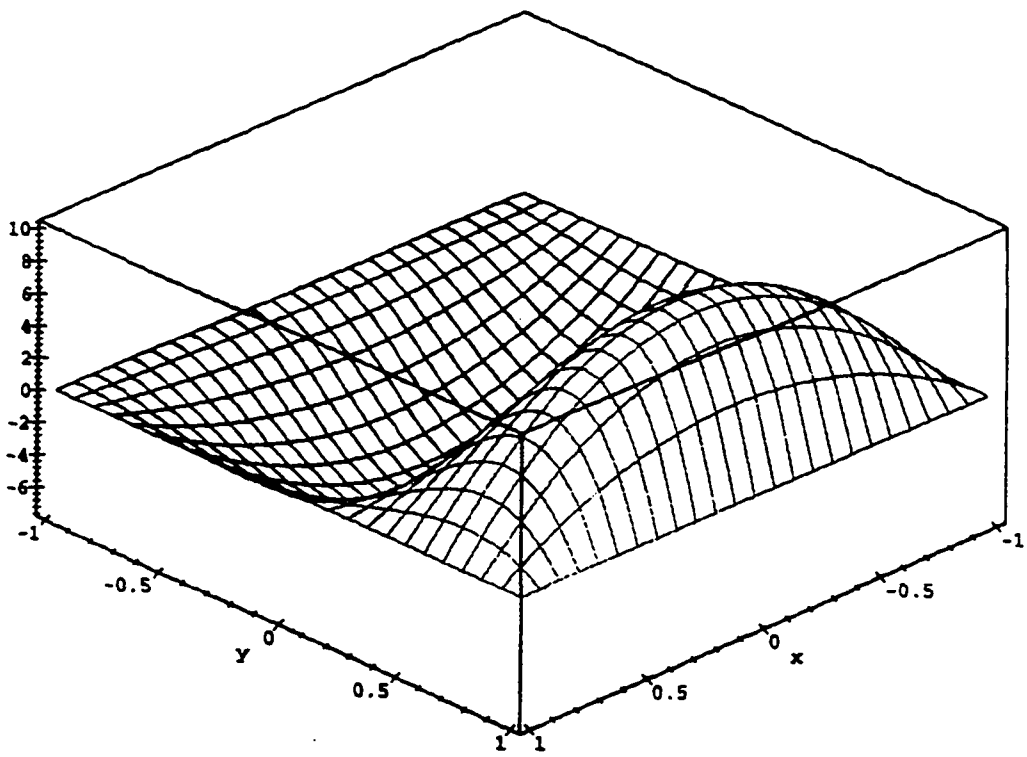

Figure I.16: Error shape due to the y-directed surface averaged current at y=b

$$\frac{-D_g}{2a}\int_{-a}^{a} dx \left.\frac{\partial\,\phi(x,y)}{\partial\,y}\right|_{y=b} \Rightarrow r_{16} = \frac{b}{64D_g}\begin{pmatrix} 24\xi - 6 + 36\xi^2 - 24\eta^2\xi + 30\eta^2\xi^4 \\ -24\xi^3 - 36\eta^2\xi^2 + 24\eta^2\xi^3 + 6\eta^2 - 30\xi^4 \end{pmatrix}$$

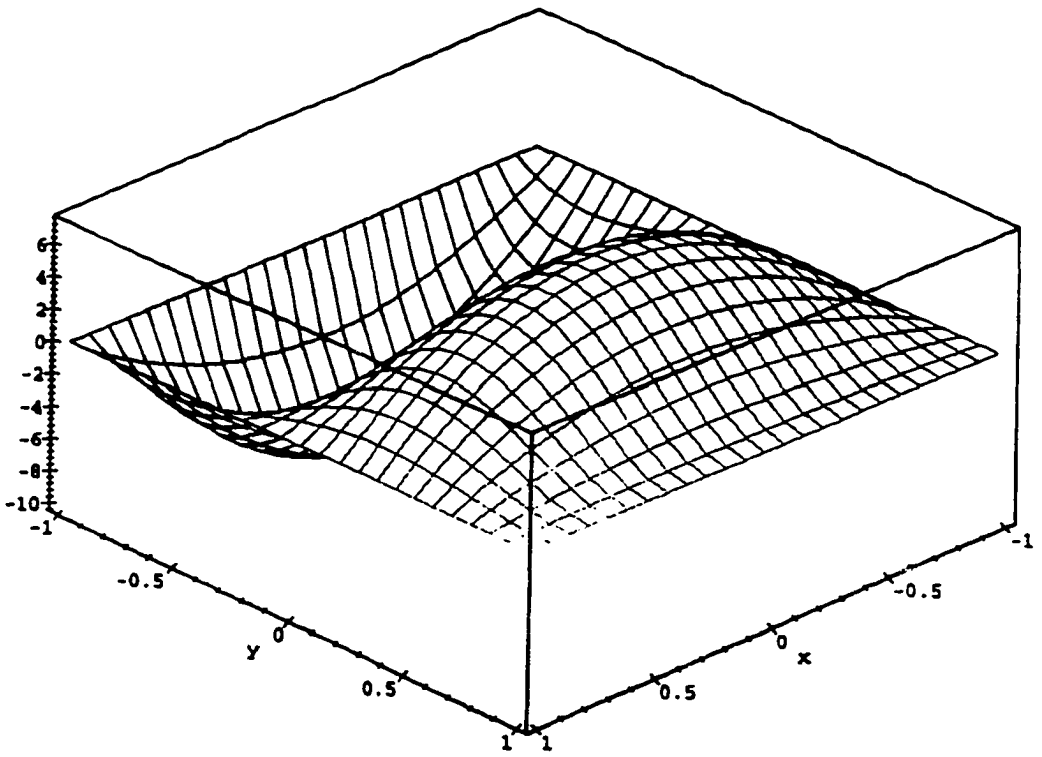

Figure I.17: Error shape due to the y-directed surface averaged current at y=-b

$$\frac{-D_g}{2a}\int_{-a}^{a} dx \left.\frac{\partial \phi(x,y)}{\partial y}\right|_{y=-b} \Rightarrow r_{17} = \frac{b}{64D_g}\begin{pmatrix} 24\xi + 6 - 36\xi^2 - 24\eta^2\xi - 30\eta^2\xi^4 \\ -24\xi^3 + 36\eta^2\xi^2 + 24\eta^2\xi^3 - 6\eta^2 + 30\xi^4 \end{pmatrix}$$

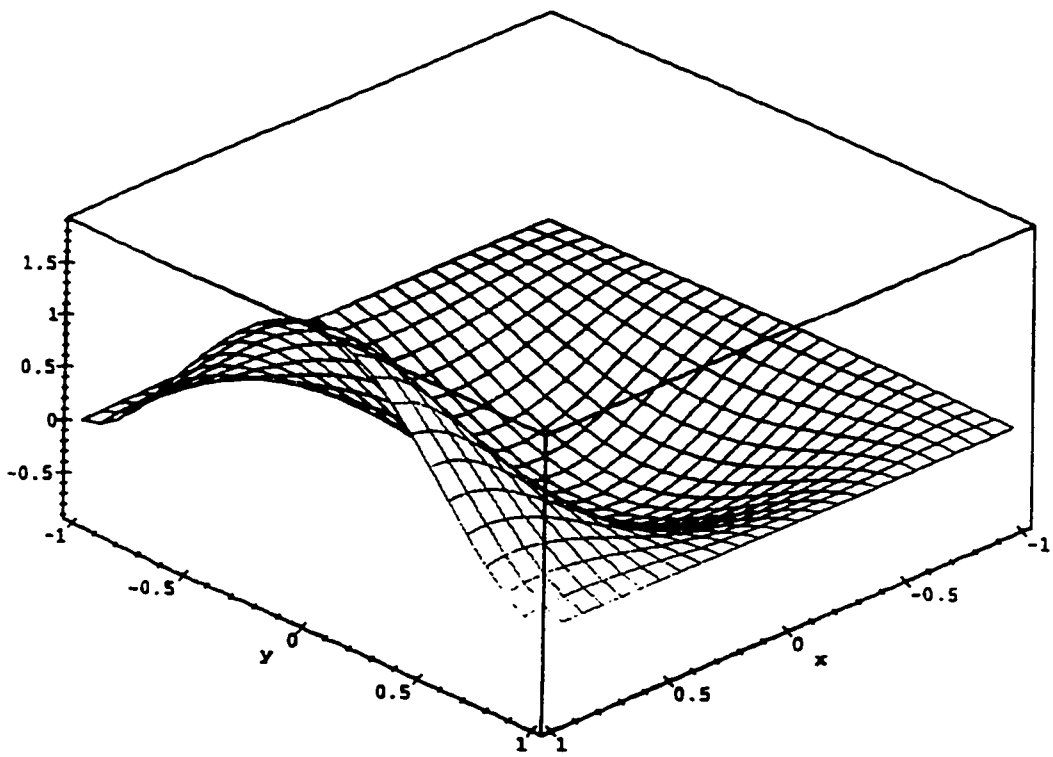

Figure I.18: Error shape due to the surface average flux at x=a

$$\frac{1}{2b}\int_{-b}^{b} dy\phi(a,x) \Rightarrow r_{18} = \frac{1}{64}\begin{pmatrix} -144\eta\xi^2 + 90\eta^4\xi^2 - 90\eta^4 + 84\eta + 90\eta^2\xi^4 + 24\eta^3\xi^2 + 60\eta\xi^4 \\ +198\eta^2 - 24\eta^3 - 48 + 78\xi^2 - 288\eta^2\xi^2 - 30\xi^4 \end{pmatrix}$$

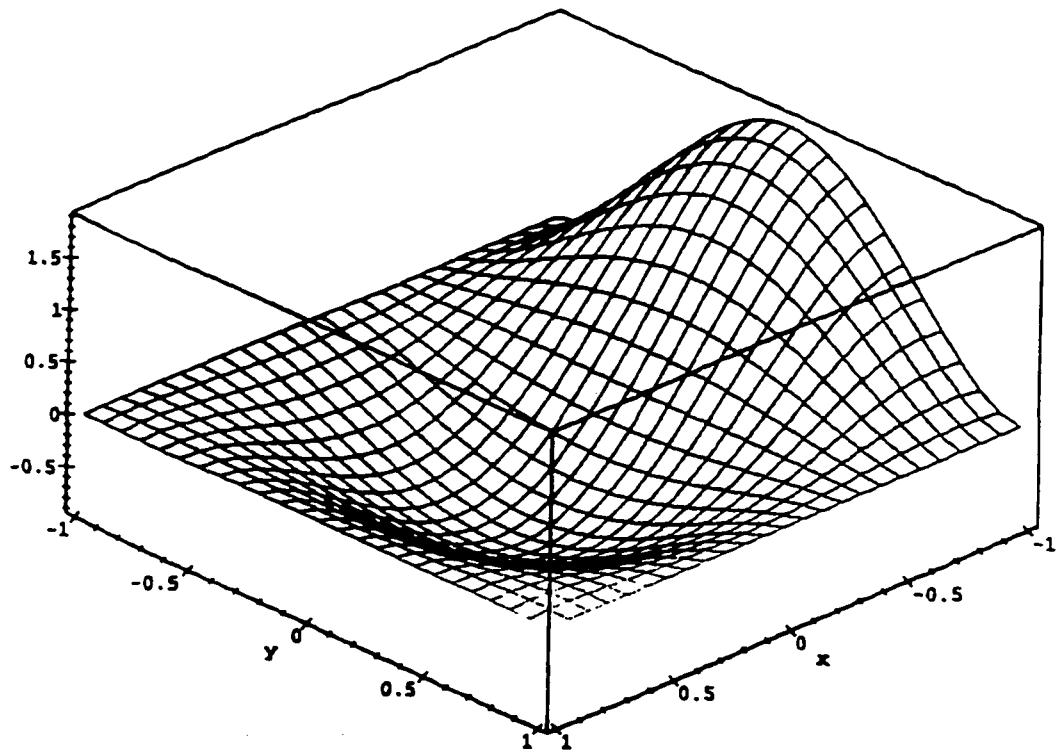

Figure I.19: Error shape due to the surface average flux at x=-a

$$\frac{1}{2b}\int_{-b}^{b} dy\phi(-a,x) \Rightarrow r_{19} = \frac{1}{64}\begin{pmatrix} +144\eta\xi^2 + 90\eta^4\xi^2 - 90\eta^4 - 84\eta + 90\eta^2\xi^4 - 24\eta^3\xi^2 - 60\eta\xi^4 \\ +198\eta^2 + 24\eta^3 - 48 + 78\xi^2 - 288\eta^2\xi^2 - 30\xi^4 \end{pmatrix}$$

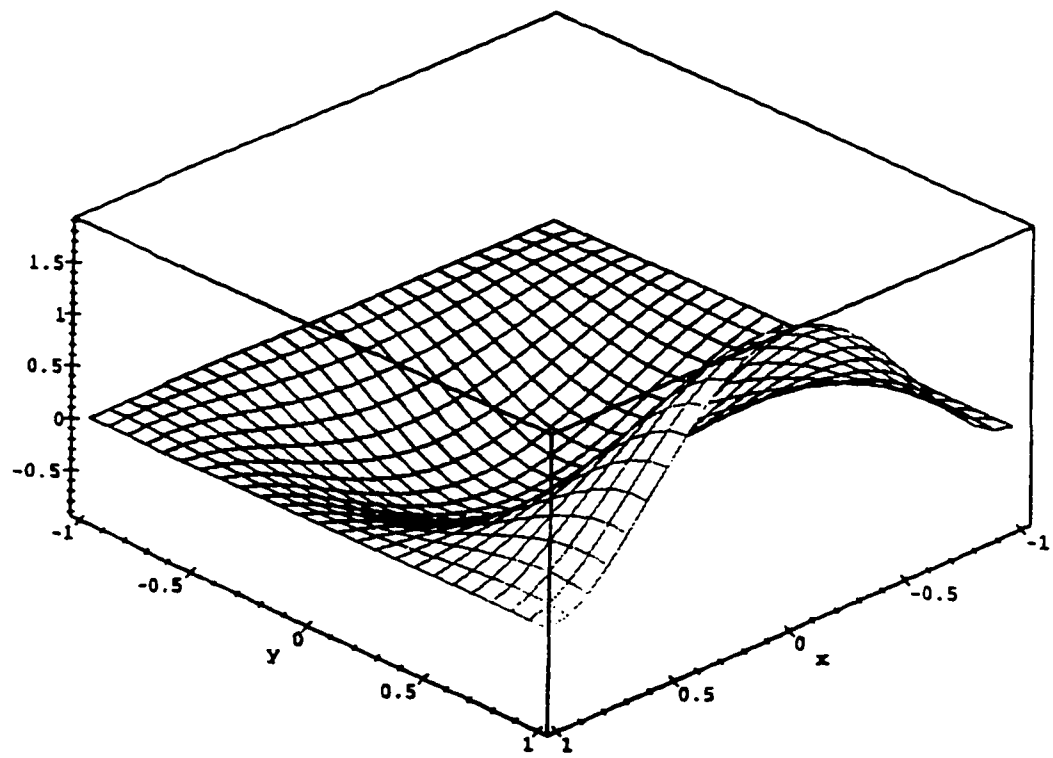

Figure I.20: Error shape due to the surface average flux at y=b

$$\frac{1}{2a}\int_{-a}^{a} dx\phi(x,b) \Rightarrow r_{20} = \frac{1}{64}\begin{pmatrix} -144\eta^2\xi + 90\eta^4\xi^2 - 90\xi^4 + 84\xi + 90\eta^2\xi^4 + 24\eta^2\xi^3 + 60\eta^4\xi \\ +198\xi^2 - 24\xi^3 - 48 + 78\eta^2 - 288\eta^2\xi^2 - 30\eta^4 \end{pmatrix}$$

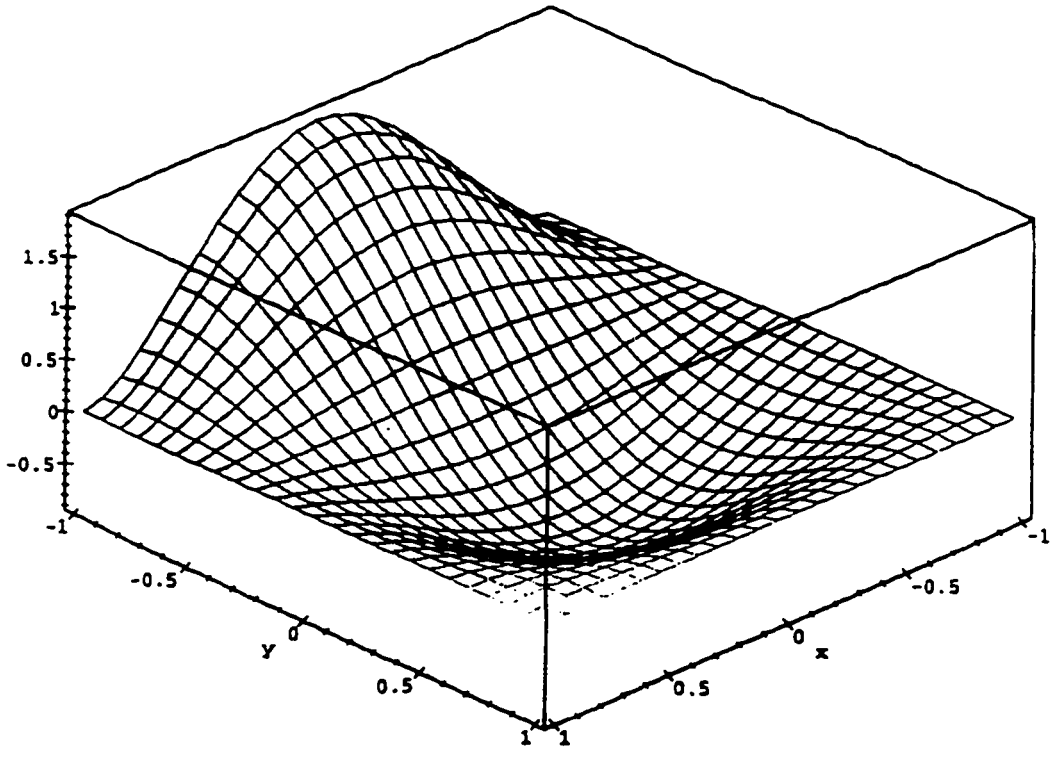

Figure I.21: Error shape due to the surface average flux at y=-b

$$\frac{1}{2a}\int_{-a}^{a} dx\phi(x,-b) \Rightarrow r_{21} = \frac{1}{64}\begin{pmatrix} 144\eta^2\xi + 90\eta^4\xi^2 - 90\xi^4 - 84\xi + 90\eta^2\xi^4 - 24\eta^2\xi^3 - 60\eta^4\xi \\ +198\xi^2 + 24\xi^3 - 48 + 78\eta^2 - 288\eta^2\xi^2 - 30\eta^4 \end{pmatrix}$$

Appendix II: Explicit Representation of the First Derivative of the Reconstructed Flux

The following are the results from an MAPLE session where the first derivative were symbolically calculated. The term, dxr, refers to the first derivative of the reconstructed flux with respect to x and the term, dyr, refers to the first derivative of the reconstructed flux with respect to y. The twenty one values of R_i were defined in Chapter 3.

```
> dxr :=simplify(dxr);

                4 4          3       4                4 2
dxr := - 1/32 (2 b  a  R7 - 360 x  Dg R1 b  + 180 x Dg R1 y  a

         3        2 2              4 2              2 2 2
  + 360 x  Dg R1 y  b  + 396 x Dg R1 b  a  - 576 x Dg R1 y  b  a

               3  2            2  3             3 2 2
  + 12 Dg R2 x y  b a  + 12 Dg R2 x  y b  a + 90 Dg R2 x  y  b

              2 2 2             3 3            3  3
  + 18 Dg R2 x  y  b  a - 16 Dg R2 y b  a  + 60 Dg R2 x  y b


                3 2           2 2 3           3  3           3 4
  - 72 Dg R2 x y b  a  - 36 Dg R2 y  b  a  + 4 Dg R2 y  b a  - 30 Dg R2 x  b

              4 3           4 3             4 2           2 4
  + 15 Dg R2 y  a  + 9 Dg R2 b  a  + 39 Dg R2 x b  a  - 6 Dg R2 x  b  a

                2 2 2             4 2           4 3
  - 144 Dg R2 x y  b  a  + 45 Dg R2 x y  a  - 15 Dg R3 y  a

            2 4               3 3           2  3
  + 6 Dg R3 x  b  a + 16 Dg R3 y b  a  - 12 Dg R3 x  y b  a

              3  3           2 2 3           3 4
  + 60 Dg R3 x  y b  + 36 Dg R3 y  b  a  - 30 Dg R3 x  b

                3 2             4 2           2 2 2
  - 72 Dg R3 x y b  a  + 39 Dg R3 x b  a  - 18 Dg R3 x  y  b  a
```

```
          4 3              3 2 2              2 2 2
- 9 Dg R3 b  a  + 90 Dg R3 x  y  b  - 144 Dg R3 x y  b  a

          3  3              4 2               3  2     4 4
- 4 Dg R3 y  b a  + 45 Dg R3 x y  a  + 12 Dg R3 x y  b a  + 2 b  a  R6

    4 4        4 4       3 5       3 5               4 3
+ 2 b  a  R8 + 2 b  a  R9 + a  b  R10 - a  b  R11 - 15 Dg R4 y  a

          2 4              3 4              4 3              3  3
+ 6 Dg R4 x  b  a - 30 Dg R4 x  b  - 9 Dg R4 b  a  - 60 Dg R4 x  y b

             3 2              3 3              3  3
+ 72 Dg R4 x y b  a  - 16 Dg R4 y b  a  + 4 Dg R4 y  b a

            2  3              3 2 2              4 2
+ 12 Dg R4 x  y b  a + 90 Dg R4 x  y  b  + 39 Dg R4 x b  a

             2 2 2              3  2              4 2
- 144 Dg R4 x y  b  a  - 12 Dg R4 x y  b a  + 45 Dg R4 x y  a


          2 2 2              2 2 3              4 3
- 18 Dg R4 x  y  b  a + 36 Dg R4 y  b  a  + 15 Dg R5 y  a

          2 4              2  3              3 2 2
- 6 Dg R5 x  b  a - 12 Dg R5 x  y b  a + 90 Dg R5 x  y  b

          3  3              4 2              4 3              3 4
- 60 Dg R5 x  y b  + 39 Dg R5 x b  a  + 9 Dg R5 b  a  - 30 Dg R5 x  b

             3 2              3  2              4 2
+ 72 Dg R5 x y b  a  - 12 Dg R5 x y  b a  + 45 Dg R5 x y  a


          3 3              2 2 2              2 2 2
+ 16 Dg R5 y b  a  - 144 Dg R5 x y  b  a  + 18 Dg R5 x  y  b  a

          2 2 3              3  3       3       3       3 4
- 36 Dg R5 y  b  a  - 4 Dg R5 y  b a  + 20 b  a R6 x  y - 4 b  a  R6 y

      2       3 2      3 2      2       4 3         4 2     2
+ 30 b  a R6 x  y  + 12 b  a  R6 x  y + 6 b  a  R6 x - 6 b  a  R6 x
```

4 3 3 3 2 4 2 2 3 2
-10 b a R6 x -12 b a R6 x y -6 b a R6 y -18 b a R6 x y

2 2 2 2 3 3 3 3 4 2 2
+18 b a R6 x y +12 b a R7 x y -20 b a R7 x y -6 b a R7 x

2 3 2 4 3 3 4 3 2 2
-30 b a R7 x y +10 b a R7 x -4 b a R7 y +12 b a R7 x y

2 2 2 2 4 3 2 4 2 2 3 2
+18 b a R7 x y -6 b a R7 x -6 b a R7 y +18 b a R7 x y

3 3 3 3 3 2 2 2 3 2
-12 b a R8 x y +20 b a R8 x y -12 b a R8 x y +18 b a R8 x y

4 2 2 2 3 2 2 4 2 3 4
-6 b a R8 x -30 b a R8 x y -6 b a R8 y +4 b a R8 y

4 3 4 3 2 2 2 2 2 4 2
+10 b a R8 x -6 b a R8 x +18 b a R8 x y -6 b a R9 y

3 2 2 4 3 2 3 2 3 4
-12 b a R9 x y +6 b a R9 x -18 b a R9 x y +4 b a R9 y

3 3 3 3 4 2 2 2 3 2
+12 b a R9 x y -20 b a R9 x y -6 b a R9 x +30 b a R9 x y

4 3 2 2 2 2 3 4 2 3 2
-10 b a R9 x +18 b a R9 x y +5 a b R10 y -18 a b R10 x y

3 3 2 2 4 2 2 3 3 4
-6 a b R10 y -12 a b R10 x y +12 a b R10 x y -4 a b R10 y

2 4 2 5 3 2 3 3 4
+15 a b R10 x y +3 a b R10 x +4 a b R10 y -5 a b R11 y

3 3 2 2 3 2 2 2 3 2 5
+6 a b R11 y -18 a b R11 x y +12 a b R11 x y +3 a b R11 x

3 4 2 4 3 2 3 2 4
+4 a b R11 y +15 a b R11 x y -4 a b R11 y -12 a b R11 x y

```
   3 5       3 5         4 4         4 4         2         4
+a b R12 - a b R13 - 12 b a R14 - 12 b a R15 - 15 a b R12 x y

     2 3       2     3 3      2     3       4     3 4
+ 18 a b R12 x y - 6 a b R12 y + 5 a b R12 y + 4 a b R12 y

      2 2       3     2 5          2 4           3 2     3
+ 12 a b R12 x y - 3 a b R12 x - 12 a b R12 x y - 4 a b R12 y

     2         4       2 3       2       2 2       3     3 4
- 15 a b R13 x y + 18 a b R13 x y + 12 a b R13 x y - 4 a b R13 y

     3 3      2     3 2      3     2 5          3       4
+ 6 a b R13 y + 4 a b R13 y - 3 a b R13 x - 5 a b R13 y

     2 4             4 3           2 4      2      2        3 2
- 12 a b R13 x y - 36 b a R14 x + 12 b a R14 y - 60 b a R14 x y

      4 2      2       2 3       2     2 2      2 2      4       3
+ 36 b a R14 x + 36 b a R14 x y - 36 b a R14 x y + 60 b a R14 x

     4 3           2        3 2      2 4      2      4 2      2
+ 36 b a R15 x + 60 b a R15 x y + 12 b a R15 y + 36 b a R15 x

     2 3       2     4       3      2 2      2 2
- 36 b a R15 x y - 60 b a R15 x - 36 b a R15 x y

       2 2      3        2      4       3 2      2      5 2
- 24 x b a R16 y - 30 x b a R16 y + 36 x b a R16 y - 6 x b a R16

       4 2              2      4       2 2      3
+ 24 x b a R16 y + 30 x b a R17 y - 24 x b a R17 y

       4 2             3 2      2      5 2
+ 24 x b a R17 y - 36 x b a R17 y + 6 x b a R17

             3 2 2              2 2 2              4 2
- 180 Dg R18 x y b + 288 Dg R18 x y b a - 90 Dg R18 x y a

           4 3             2 2 2             4 3
- 42 Dg R18 b a - 36 Dg R18 x y b a - 30 Dg R18 y a

             4 2             2 4              2 2 3
- 198 Dg R18 x b a + 36 Dg R18 x b a + 72 Dg R18 y b a
```

```
                3 4              2 2 2                    4 2
+ 180 Dg R18 x  b  + 36 Dg R19 x  y  b  a - 198 Dg R19 x b  a

              4 3              3 2 2                3 4
+ 42 Dg R19 b  a  - 180 Dg R19 x  y  b  + 180 Dg R19 x  b

              2 4                4 2              4 3
- 36 Dg R19 x  b  a - 90 Dg R19 x y  a  + 30 Dg R19 y  a

                2 2 2              2 2 3       3         2 2
+ 288 Dg R19 x y  b  a  - 72 Dg R19 y  b  a  - 180 x  Dg R20 y  b

      3         4                  3 2                3   2
+ 60 x  Dg R20 b  + 144 x Dg R20 y b  a  - 24 x Dg R20 y  b a

                2 2 2       3          3                4 2
+ 288 x Dg R20 y  b  a  - 120 x  Dg R20 y b  - 90 x Dg R20 y  a

              4 2                3 2        3          2 2
- 78 x Dg R20 b  a  - 144 x Dg R21 y b  a  - 180 x  Dg R21 y  b

              4 2                3   2        3          3
- 90 x Dg R21 y  a  + 24 x Dg R21 y  b a  + 120 x  Dg R21 y b

              4 2                2 2 2      3          4
- 78 x Dg R21 b  a  + 288 x Dg R21 y  b  a  + 60 x  Dg R21 b )

 /  4 4
 / (b  a  Dg)
/
```

(Eq. II.1a)

```
> dyr :=simplify(dyr);

                4 4          4 4          4 4        3 5       3 5
dyr := - 1/32 (2 a  b  R13 + 2 a  b  R12 + 2 a  b  R11 + b  a  R6 - b  a  R7
```

2 4 3 4 3 2 2
+ 396 y Dg R1 b a - 360 y Dg R1 a + 360 y Dg R1 x a

4 2 2 2 2 2 2
+ 180 y Dg R1 x b - 576 y Dg R1 x b a + 18 Dg R2 x y b a

3 3 4 2 3 4 3 2
+ 4 Dg R2 x b a + 45 Dg R2 x y b + 9 Dg R2 b a + 12 Dg R2 x y b a

3 4 3 3 4 3 2 3 2
- 30 Dg R2 y a - 16 Dg R2 x b a + 15 Dg R2 x b - 36 Dg R2 x b a

2 3 2 3 3 3
- 72 Dg R2 x y b a + 12 Dg R2 x y b a + 60 Dg R2 x y a

2 4 2 2 2 2 4
+ 39 Dg R2 y b a - 144 Dg R2 x y b a - 6 Dg R2 y b a

2 3 2 3 3 2 4
+ 90 Dg R2 x y a - 60 Dg R3 x y a + 39 Dg R3 y b a

3 3 2 4 3 3 4 3
+ 16 Dg R3 x b a - 6 Dg R3 y b a - 4 Dg R3 x b a + 15 Dg R3 x b

2 3 2 3 2 3 4 3 4
+ 72 Dg R3 x y b a - 36 Dg R3 x b a - 30 Dg R3 y a + 9 Dg R3 b a

3 2 4 2 2 2 2
- 12 Dg R3 x y b a + 45 Dg R3 x y b - 144 Dg R3 x y b a

2 3 2 3 2 2 2 2
- 12 Dg R3 x y b a + 90 Dg R3 x y a + 18 Dg R3 x y b a

2 4 3 3 3 4 2 4
+ 39 Dg R4 y b a - 60 Dg R4 x y a - 9 Dg R4 b a + 6 Dg R4 y b a

4 3 2 3 2 3 3
- 15 Dg R4 x b + 36 Dg R4 x b a - 16 Dg R4 x b a

2 3 3 3 3 4 4 2
+ 12 Dg R4 x y b a + 4 Dg R4 x b a - 30 Dg R4 y a + 45 Dg R4 x y b

```
          2  2 2          2 2  2          2 3 2
-144 Dg R4 x  y b  a  -18 Dg R4 x  y  b a  +90 Dg R4 x  y  a

          3  2          2 3          3 3
-12 Dg R4 x  y b  a +72 Dg R4 x y b  a  +60 Dg R5 x y  a

          3 4          3 3          2  4          4  2
-9 Dg R5 b  a  -4 Dg R5 x  b  a +6 Dg R5 y  b a  +45 Dg R5 x  y b

          2 4          4 3          3 4          2 3 2
+39 Dg R5 y b  a  -15 Dg R5 x  b  -30 Dg R5 y  a  +36 Dg R5 x  b  a

          2 2  2          2 3 2          3 3
-18 Dg R5 x  y  b a  +90 Dg R5 x  y  a  +16 Dg R5 x b  a

          2  2 2          3  2          2 3
-144 Dg R5 x  y b  a  +12 Dg R5 x  y b  a -72 Dg R5 x y b  a

          2  3    3    4    3 4          2    4
-12 Dg R5 x y  b a  +5 b  a R6 x  -4 b  a  R6 x +15 b  a R6 x  y

    3 2    3    2 5        3 3    2    2 4
+4 b  a  R6 x  +3 b  a  R6 y -6 b  a  R6 x  -12 b  a  R6 x y

    2 3    2      2 2    3    3 3    2    3    4
-18 b  a  R6 x  y +12 b  a  R6 x  y +6 b  a  R7 x  -5 b  a R7 x

    2    4    3 4        3 2    3    2 2    3
-15 b  a R7 x  y -4 b  a  R7 x +4 b  a  R7 x  +12 b  a  R7 x  y

   2 5        2 4          2 3    2    3 5    3 5
-3 b  a  R7 y -12 b  a  R7 x y +18 b  a  R7 x  y +b  a  R8 -b  a  R9

   4 4        4 4        4 4        3 3    2    3    4
+2 a  b  R10 -12 a  b  R16 -12 a  b  R17 -6 b  a  R8 x  +5 b  a R8 x

   3 2    3    2 3    2    2 5        2    4
-4 b  a  R8 x  +18 b  a  R8 x  y -3 b  a  R8 y -15 b  a R8 x  y

    2 4          3 4          2 2    3    2 4
-12 b  a  R8 x y +4 b  a  R8 x +12 b  a  R8 x  y -12 b  a  R9 x y

   3 2    3    2 3    2    3 4        3 3    2
-4 b  a  R9 x  -18 b  a  R9 x  y +4 b  a  R9 x +6 b  a  R9 x
```

3 4 2 4 2 5 2 2 3
-5 b a R9 x + 15 b a R9 x y + 3 b a R9 y + 12 b a R9 x y

3 3 2 3 2 3 3 2 4 2
+20 a b R10 x y - 18 a b R10 x y - 12 a b R10 x y - 6 a b R10 x

2 2 2 2 4 2 2 3 4 2 2 3
+18 a b R10 x y - 6 a b R10 y - 4 a b R10 x + 30 a b R10 x y

4 3 3 2 2 4 3 3 3
-10 a b R10 y + 12 a b R10 x y + 6 a b R10 y - 20 a b R11 x y

3 3 2 3 2 2 2 2 3 4
+12 a b R11 x y - 18 a b R11 x y + 18 a b R11 x y + 4 a b R11 x

4 2 2 4 3 2 2 3 3 2 2
-6 a b R11 y - 10 a b R11 y + 30 a b R11 x y - 12 a b R11 x y

4 3 2 4 2 2 2 3 2 3 2
+6 a b R11 y - 6 a b R11 x - 30 a b R12 x y + 18 a b R12 x y

3 3 3 3 3 4 2 2 2 2
-12 a b R12 x y + 20 a b R12 x y + 4 a b R12 x + 18 a b R12 x y

4 3 4 2 2 4 3 2 4 2
+10 a b R12 y - 6 a b R12 y - 6 a b R12 y - 6 a b R12 x

3 2 2 2 2 3 2 3 2
-12 a b R12 x y - 30 a b R13 x y + 18 a b R13 x y

2 2 2 2 3 4 4 3 3 3
+18 a b R13 x y - 4 a b R13 x - 6 a b R13 y + 12 a b R13 x y

4 2 2 3 2 2 4 3 3 3
-6 a b R13 y + 12 a b R13 x y + 10 a b R13 y - 20 a b R13 x y

2 4 2 2 4 2 4 2 3 2
-6 a b R13 x + 24 y b a R14 x - 30 y b a R14 x + 36 y b a R14 x

2 2 3 2 5 2 5 2 4
-24 y b a R14 x - 6 y b a R14 + 6 y b a R15 + 30 y b a R15 x

```
      2 4             2 3      2       2 2      3
+24 y b  a  R15 x - 36 y b  a  R15 x - 24 y b  a  R15 x

     2 2      2 2      2       2 3      2 3      2
- 36 a  b  R16 x  y - 60 a  b R16 x  y + 36 a  b  R16 x  y


     4        3      4 2      2      2 4      2      4 3
+ 60 a  b R16 y  + 36 a  b  R16 y  + 12 a  b  R16 x  - 36 a  b  R16 y

     4 2      2      4       3      2       2 3      4 3
+ 36 a  b  R17 y  - 60 a  b R17 y  + 60 a  b R17 x  y + 36 a  b  R17 y


     2 2      2 2      2 4      2      2 3      2
- 36 a  b  R17 x  y + 12 a  b  R17 x  - 36 a  b  R17 x  y

              4 2                2 2 2      3        4
- 90 y Dg R18 x  b  + 288 y Dg R18 x  b  a  + 60 y  Dg R18 a

      3         2 2              3 2        3          3
- 180 y  Dg R18 x  a  - 24 y Dg R18 x  b  a - 120 y  Dg R18 x a

              2 4                2 3       3        4
- 78 y Dg R18 b  a  + 144 y Dg R18 x b  a  + 60 y  Dg R19 a

              3 2                4 2       3        2 2
+ 24 y Dg R19 x  b  a - 90 y Dg R19 x  b  - 180 y  Dg R19 x  a

       3          3              2 4                2 2 2
+ 120 y  Dg R19 x a  - 78 y Dg R19 b  a  + 288 y Dg R19 x  b  a

               2 3              4  2             2 3 2
- 144 y Dg R19 x b  a  - 90 Dg R20 x  y b  + 72 Dg R20 x  b  a

             2  4              3 4              2 2  2
+ 36 Dg R20 y  b a  + 180 Dg R20 y  a  - 36 Dg R20 x  y  b a

              2  2 2             3 4               2 4
+ 288 Dg R20 x  y b  a  - 42 Dg R20 b  a  - 198 Dg R20 y b  a

            4 3              2 3 2            2 3 2
- 30 Dg R20 x  b  - 180 Dg R20 x  y  a  - 72 Dg R21 x  b  a
```

$$
\begin{aligned}
&\overset{4\ \ 2}{\text{- 90 Dg R21 x y b}} + \overset{3\ 4}{\text{180 Dg R21 y a}} + \overset{3\ 4}{\text{42 Dg R21 b a}} \\
&\overset{2\ 3\ 2}{\text{- 180 Dg R21 x y a}} + \overset{2\ 2\ \ 2}{\text{36 Dg R21 x y b a}} + \overset{4\ 3}{\text{30 Dg R21 x b}} \\
&\overset{2\ 4}{\text{- 198 Dg R21 y b a}} - \overset{2\ \ 4}{\text{36 Dg R21 y b a}} + \overset{2\ \ 2\ 2\ /\ \ 4\ 4}{\text{288 Dg R21 x y b a}} \text{)// (b a Dg)}
\end{aligned}
$$

(Eq. II.2b)

References

1. Numerical Mathematics, Gunther Hammerlin and Karl-Heinz Hoffmann, Springer-Verlag, New York, 1991, p.206.

2. "The Determination of the Pin Power Distribution in a Reactor on the Basis of Nodal Coarse Mesh Calculations", Klaus Koebke and Manfred R. Wagner, Atomkernenergie, Bd. 30(1977), Lfg. 2, pp. 136-142.

3. "Verification of the SIMULATE-3 Pin Power Distribution Calculation", A.S. DiGiovine, J.P. Gorski, and M.A. Tremblay, Nuclear Science and Engineering, Vol. 103, 1989, pp. 420-426.

4. Computational Methods of Neutron Transport, E.E. Lewis and W.F. Miller, Jr., John Wiley & Sons, New York, 1984, p. 42.

5. Nuclear Reactor Analysis, James J. Duderstadt and Louis J. Hamilton, John Wiley & Sons, New York, 1976.

6. Development of Advanced Nodal Diffusion Methods for Modern Computer Architectures, Hrabri Luka Rajic, PhD thesis, University of Illinois, Department of Nuclear Engineering, 1988, p. 26.

7. "Assembly Homogenization Techniques for Light Water Reactor Analysis", K.S. Smith, Progress in Nuclear Energy, Vol. 17, No. 3, pp. 303-335.

8. Introduction to Nuclear Engineering, John R. Lamarsh, Addison-Wesley Publishing Company, Reading, MA, 1977, p. 231.

9. Nuclear Reactor Theory, Allan Henry, MIT Press, Boston, MA

10. Introduction to Numerical Analysis, F.B. Hildebrand, Dover Publications, Inc., New York, 1974, p.81.

11. Interpolation and Approximation, Phillip J. Davis, Dover Publishing, Inc., New York, 1963.

12. "A Bibliography of Multivariate Approximation", Richard Franke and Larry Schumaker, Topics in Multivariate Approximation, edited by C.K. Chui, L.L. Schumaker, and F.I. Utreras, Academic Press, Boston, 1987, pp. 275-335.

13. "Reconstruction of Fuel Pin Powers from Nodal Results", H.S. Khalil, P.J. Finck, and A.F. Henry, Advances in Reactor Computation, Vol. 1, Salt Lake City, UT, March 28-31, 1983, p. 367-379.

14. "Determination of Local Pin Powers in the Framework of Nodal Coarse-Mesh Solutions", Flemming Nissen, Advances in Reactor Computations, Vol. 1, Salt Lake City, UT, March 28-31, 1983, pp. 380-395.

15. "SIMULATE-3 Pin Power Reconstruction: Methodology and Benchmarking", K.R. Rempe and K.S. Smith, Nuclear Science and Engineering, Vol. 103, 1989, pp. 334-342.

16. MAPLE V, registered trademark of Waterloo Maple Software, 1993.

17. "On the Reconstruction of Local Homogeneous Neutron Flux and Current Distributions of Light Water Reactors from Nodal Schemes", K. Koebke and L. Hetzelt, Nuclear Science and Engineering, Vol. 91, 1985, pp. 123-131.

18. "A Formula for Node Edge Fluxes in Rectangular Geometry", Sten-Orjan Lindahl, International Conference on the Physics of Reactors: Operation, Design

and Computation, Vol. 2, April 23-27, 1990, Marseille, France, pp. IV-56 to IV-63.

19. Theory of Approximation, N.I. Achieser, Dover Publications, Inc., New York, 1992.

20. Richard Holm, Reactor Supervisor, University of Illinois Advanced TRIGA Reactor, personel communications, 1996.

21. The Mathematics of Nonlinear Programming, Peresinni, Sullivan, and Uhl, John Wiley & Sons, New York, 1992.

22. Manual for the RRAP program created by Atoms Analysis, Oregon, 1994.

23. Manual for MCNP programs created by Los Alamos National Laboratory, Los Alamos, New Mexico.

24. Safety Analysis Report for the University of Illinois Advanced TRIGA Reactor, University of Illinois, 1959.

25. Facility Records for the University of Illinois Advanced TRIGA reactor, 1959 to present.

26. Two Group Reactor Theory, J. Lawrence Meem, Gordon and Breach, New York, 1964.

Vita

Brian Mark Golchert was born on 17 June 1961. He completed high school at Downers Grove Community High School North in June of 1979. He then attended the University of Illinois, majoring in nuclear engineering. He obtained his bachelor's degree in January of 1984. After studying physics for over two years, he was admitted to the graduate college at the University of Illinois, again majoring in nuclear engineering. He obtained his master's degree in January of 1989 and is now completing his doctorate in the same field. He is happily married to his wife, Lucia, and they live with their lovely daughter, Ana, in Peoria, Illinois.